FOR FURTHER TRAN

USARTL-TR-78-2B

AD A054827

HELICOPTER TRANSMISSION VIBRATION AND NOISE REDUCTION PROGRAM

Volume II - User's Manual

John J. Sciarra, Robert W. Howells, Joseph W. Lenski, Jr., Raymond J. Drago
Boeing Vertol Company
P.O. Box 16858
Philadelphia, Pa. 19142

March 1978

Final Report for Period June 1974 - October 1977

Approved for public release; distribution unlimited.

78 06 08 035

Prepared for

APPLIED TECHNOLOGY LABORATORY
U. S. ARMY RESEARCH AND TECHNOLOGY LABORATORIES (AVRADCOM)
Fort Eustis, Va. 23604

U. S. ARMY MATERIALS AND MECHANICS RESEARCH CENTER
WATERTOWN, MASS. 02172

APPLIED TECHNOLOGY LABORATORY POSITION STATEMENT

This report provides the details of the eight computer programs that comprise the analytical tool developed under this effort. These computer programs have been recorded on tape and are available, upon application, through the Applied Technology Laboratory on a reimbursable basis.

Mr. Allen C. Royal of the Propulsion Technical Area, Technology Applications Division, served as project engineer for this effort.

DISCLAIMERS

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission, to manufacture, use, or sell any patented invention that may in any way be related thereto.

Trade names cited in this report do not constitute an official endorsement or approval of the use of such commercial hardware or software.

DISPOSITION INSTRUCTIONS

Destroy this report when no longer needed. Do not return it to the originator.

DISCLAIMER NOTICE

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: USARTL-TR-78-2B

2. GOVT ACCESSION NO.:

3. RECIPIENT'S CATALOG NUMBER:

4. TITLE (and Subtitle): HELICOPTER TRANSMISSION VIBRATION AND NOISE REDUCTION PROGRAM. Volume II. User's Manual.

5. TYPE OF REPORT & PERIOD COVERED: Final Report. Jun[redacted]74 - Oct[redacted]77

6. PERFORMING ORG. REPORT NUMBER: D210-11236-2

7. AUTHOR(s): John J. Sciarra, Joseph W. Lenski, Jr., Robert W. Howells, Raymond J. Drago

8. CONTRACT OR GRANT NUMBER(s): DAAJ02-74-C-0040

9. PERFORMING ORGANIZATION NAME AND ADDRESS:
Boeing Vertol Company
P.O. Box 16858
Philadelphia, PA 19142

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62207A 1G262207AH89 02 004 EK

11. CONTROLLING OFFICE NAME AND ADDRESS:
Applied Technology Laboratory
US Army Research and Technology Laboratories (AVRADCOM), Fort Eustis, VA

12. REPORT DATE: Mar[redacted]78

13. NUMBER OF PAGES: 429

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):
U.S. Army Materials and Mechanics Research Center
Watertown, Massachusetts

15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

12 431p

16. DISTRIBUTION STATEMENT (of this Report):

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report):

18. SUPPLEMENTARY NOTES:

Volume II of a 2-volume report

19. KEY WORDS (Continue on reverse side if necessary and identify by block number):

Helicopter transmission
Vibration/noise reduction
Analytical tools for prediction
Trade studies
Dynamically quiet
Design stage
Computer-oriented
Computer programs
Use and application
Reduced vibration/noise
Longer life

20. ABSTRACT (Continue on reverse side if necessary and identify by block number):

The objective of the Helicopter Transmission Vibration/Noise Reduction Program was to generate analytical tools for the prediction and reduction of helicopter transmission vibration/noise that provide the capability to perform trade studies during the design stage of a program. Application of this optimization capability yields drive train components that are dynamically quiet with reduced vibration/noise levels and inherently longer

403 612

20. (continued)

life. The work conducted under this program is highly computer-oriented and makes extensive use of several computer programs as indicated in the technical report (Volume I). This User's Manual describes these computer programs, presents rationale for their use, and discusses their application.

PREFACE

This report summarizes the results of the "Helicopter Transmission Vibration and Noise Reduction Program." The report covers the work accomplished during the 40-month period from June 1974 through October 1977 and is composed of two volumes. Volume I is the Technical Report, and Volume II is the User's Manual.

The work outlined herein has been performed under U.S. Army contract DAAJ02-74-C-0040 and under the technical cognizance of Mr. Allen Royal, U.S. Army Research and Technology Laboratories, Fort Eustis, Virginia.

This program was conducted at the Boeing Vertol Company under the technical direction of Mr. A. J. Lemanski (Program Manager), Chief of the Advanced Power Train Technology Department. Principal investigators for the program were Mr. John J. Sciarra (Project Engineer), Mr. Robert W. Howells, Mr. Joseph W. Lenski, Jr., and Mr. Raymond J. Drago.

78 06 08 0

Preceding Page Blank - NOT FILMED

TABLE OF CONTENTS

APPENDIXES

Page

LIST OF ILLUSTRATIONS

Figure		Page
22	Schematic Diagram of Torsional System Model for CH-47 Forward Transmission and Closed-Loop Test Stand	111
23	Schematic Diagram of Closed-Loop Test Stand	113
24	TORRP (R-32) Input Sheet	114
25	Sample Output for R-32 Torsional Response	127
26	Mathematical Model of CH-47 Transmission	139
27	Input Sheet - Control Numbers	141
28	Coordinates and Nodal Boundary Conditions	142
29	Structural Elements	143
30	Masses and Mass Moments of Inertia	144
31	Damped Forced Response Parameters	145
32	Exciting Loads	146
33	Input Sheet - Control Numbers	147
34	Coordinates and Nodal Boundary Conditions	149
35	Structural Elements	150
36	Masses and Mass Moments of Inertia	152
37	Damped Forced Response Parameters	153
38	Exciting Loads	154
39	Sample D-82/C-51 Output	156
40	D-82/C-51 Model of Internal Components	170
41	NASTRAN Structural Model Procedure	177
42	NASTRAN Model of Cylinder	179
43	NASTRAN CPU Time	179
44	Convergence Plot	179
45	Predicted Mode Shapes for Cylinder	180

LIST OF TABLES

Preceding Page Blank - Not Filmed

INTRODUCTION

Considerable attention has been focused in recent years on the reduction of noise levels for both military and civil helicopters. Helicopter noise emanates from three major sources - the rotor blades, engines, and transmissions. Exterior noise is dominated by the rotors and engines, although the transmission also contributes to this noise. Minimization of the exterior noise is important to reduce the annoyance to communities near civil helicopter operations and to reduce the detectable noise signature of military helicopters. The interior cabin noise is predominantly due to the transmissions, with the engines and rotors being secondary sources. Interior noise not only degrades crew performance by causing annoyance and fatigue, but interferes with reliable communication and may cause hearing damage. Comfortable interior noise levels are essential for passenger acceptance of civil helicopters.

By any of the numerous standards in existence for scaling annoyance and reactions to noise, transmission noise is particularly objectionable. Noise in excess of 120 dB has been measured for the transmission of a medium transport helicopter which, for comparison, approaches the noise level of an air-raid siren. Not only is this noise level high, but its frequency typically falls within the sensitive 1000-5000 Hz range which is particularly annoying to the human ear. Furthermore, the pure tonal content, which results in a high-pitched whine, is subjectively much more annoying than broadband noise.

Transmission noise and the inherent structural vibrations which generate this noise have been of concern to helicopter designers for many years. Until recently, analytical methods have not been available to predict and reduce transmission vibration/noise problems in advance. The conventional means of controlling transmission noise has generally been to add acoustical enclosures after the hardware is built and a noise problem has become evident. Since practical enclosures are limited in noise attenuation by unavoidable sound leaks in seams and access doors, adequate attenuation is not provided for advanced helicopter drive systems of increased power. Not only do these enclosures impose considerable weight and maintainability penalties, but they do not reduce the deleterious effect of the accompanying vibrations which contribute to material fatigue and fretting at joints. The objective of this report is to document computer programs that will provide the capability to perform trade studies during the design stage of a program. This capability will yield optimized drive train components that are dynamically quiet with inherently longer life and reduced vibration and attendant noise levels.

The following programs which were used in this contract are briefly described and sample input and output cases are provided to assist the user in applying these programs to other applications. The programs identified in this report are:

- Gear Tooth Mesh Excitation and Compliance Input Generator Program (Tregold's Approximation)
- Mesh Dynamics for High-Contact-Ratio Spur Gears (R-75)
- Program for Prediction of Gear Mesh Excitation Spectra - GGEAR (R-67)
- Dynamic Gear Tooth Force Analyses - TORRP (R-32)
- Complex Structural Dynamic Analysis Computer Program Using Stiffness Methods (D-82/C-51)
- NASA Structural Analysis Program - NASTRAN (S-70)
- Strain Energy Analysis Programs (S-68 and S-83)

BACKGROUND

The transfer of torque between mating gears is not uniform due to tooth profile errors and the elastic deformation of the gear teeth under load. This nonuniform transfer of torque produces a dynamic force at the gear mesh frequency (number of teeth x rpm) and its multiples which excites the coupled torsional/lateral vibratory modes of the gear shaft. This lateral vibration (or bending) produces displacements at the bearing locations which excite the housing and cause it to vibrate, thus radiating noise (Figure 1). Furthermore, the dynamic characteristics of the housing may magnify its displacements and the resulting noise. This transfer of mesh energy from its source to remote locations is shown in Figure 2 as measured noise levels.

Controlling the dynamic response of the transmission is a desirable approach to noise reduction, since avoidance of resonance reduces shaft deflections at the bearings and inherently increases the life of dynamic components and transmission reliability.

Computer program GGEAR/HCR (R-75) has been prepared by Mechanical Technology Incorporated (MTI) for the Boeing Vertol Company. High-contact-ratio (HCR) gearing reduces the nonuniform load transfer between gears and hence mesh noise/vibration generation.

Computer program GGFAR (R-67) (Reference 1), which was also prepared by MTI as an extension of GEARO (Reference 2), takes gear tooth geometry (driving/driven) as well as material properties to simulate mesh cycles. This computer program is used to calculate approximate gear mesh compliance (Reference 1, page 156) and harmonics of the meshing error.

Utilizing the mesh compliance and a meshing error of R-67, the gear teeth dynamic load may be obtained using TORRP (R-32). This is basically an uncoupled torsional Holtzer analysis with planetary capability (Reference 2). Other input would be shaft and gear geometry as well as exciting frequency.

---

1. Badgley, R. H., and Laskin, I., PROGRAM FOR HELICOPTER GEARBOX NOISE PREDICTION AND REDUCTION, Mechanical Technology Incorporated, USAAVLABS Technical Report 70-12, U.S. Army Aviation Materiel Laboratories, Fort Eustis, Virginia, March 1970, AD869822.

2. Laskin, I., Orcutt, F. K., and Shipley, E. E., ANALYSIS OF NOISE GENERATED BY UH-1 HELICOPTER TRANSMISSION, Mechanical Technology Incorporated, USAAVLABS Technical Report 68-41, U.S. Army Aviation Materiel Laboratories, Fort Eustis, Virginia, June 1968, AD675457.

Reduction of the dynamic excitation of the housing is accomplished by minimizing the dynamic forces at the shaft support bearings. This is a two-fold task. First, the excitation due to the dynamic tooth forces is calculated from the gear geometry and operating conditions. Second, the damped force response of the shafts responding to the tooth mesh excitation loads is calculated from a finite element model and the shaft is detuned using strain energy methods to minimize the displacement at the bearings. The development of this method,

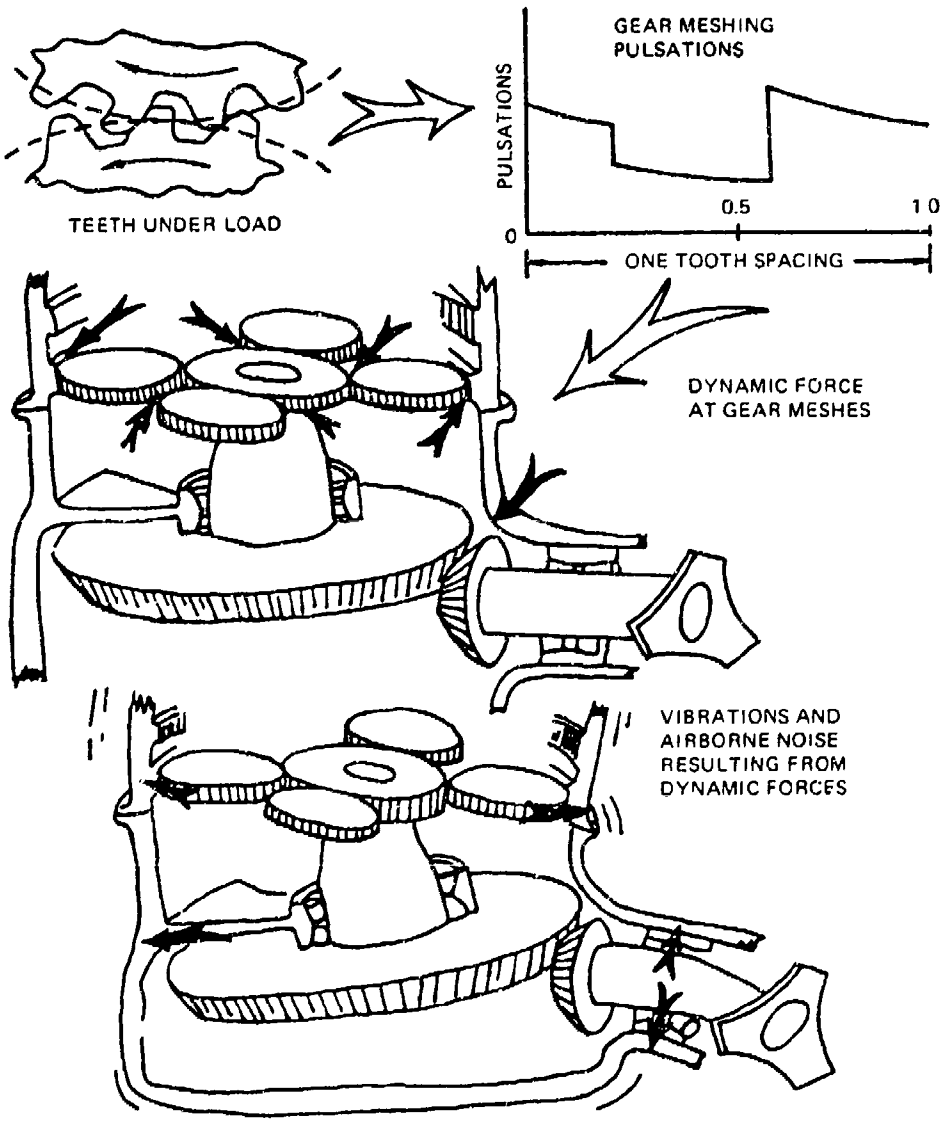


Figure 1. Sources of Transmission Noise.

accomplishment of extensive dynamic testing, and correlation of data are described fully in References 3 and 4. Finally, the dynamic forces associated with the optimum configuration of the internal components are applied to the model of the housing. To study the response of the transmission housing to these forces and to minimize the noise produced, a finite element model of the housing was developed and analyzed using NASTRAN.

MICROPHONE INSTALLATION

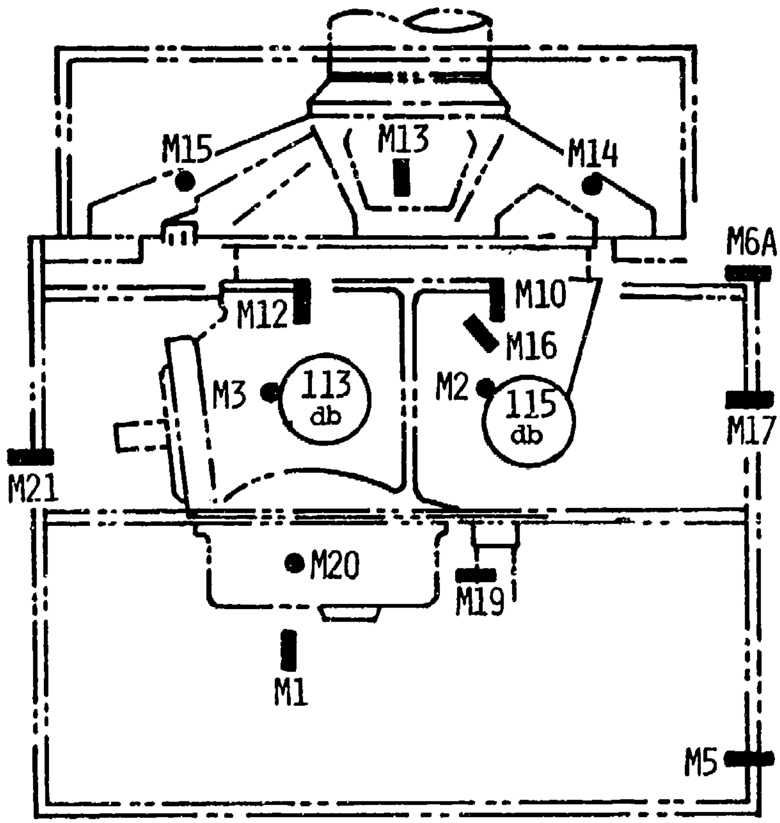


Figure 2. Maximum Measured Noise Levels (7460 RPM at 80% Torque).

---

3. Hartman, R. M., A DYNAMICS APPROACH TO HELICOPTER TRANSMISSION NOISE REDUCTION AND IMPROVED RELIABILITY, Presented at the 29th Annual National Forum of the American Helicopter Society, Washington, D.C., May 1973.

4. Hartman, R. M., and Badgley, R. H., MODEL 301 HLH/ATC TRANSMISSION NOISE REDUCTION PROGRAM, The Boeing Vertol Company, USAAMRDL Technical Report 74-58, U.S. Army Air Mobility Research and Development Laboratory, Fort Eustis, Virginia, May 1974, AD784132.

Two computer programs have been developed to calculate the strain energy/density distribution of finite element models vibrating in various modes (References 5 and 6).

The first, S-68, takes any mode shape of the internal components finite element model as determined by finite element computer program D-82 and calculates the strain energy/density distribution. D-82 calculates the coupled bending (vertical and lateral)/torsion natural frequencies and mode shapes of the internal components finite element model which includes shaft data, gear tooth mesh compliance, bearing spring rates, lower planetary representation, and sync shaft torsional spring constant.

The second strain density computer program, S-83, takes a NASTRAN modified checkpoint tape as input and calculates the strain density distribution for the transmission housing plate structural elements.

The basis for finding the strain density distribution is that it yields the optimum locations for structural modification for detuning a structure for least weight penalty.

Work conducted under this program was directed toward analyzing the Boeing Vertol CH-47 forward rotor transmission housing which is composed of three major sections: upper cover, ring gear, and case. The upper cover provides lugs for mounting the transmission to the airframe and transmits the rotor system loads. The case contains and supports the main bevel gears. The ring gear, which connects the upper cover and case, contains the planetary gear system. This natural division of the housing was adhered to for all the modeling completed during this program.

The geometric grid points for the model were defined from design drawings and by cross-checking on an actual housing. CQUAD2 (Quadrilateral) and CTRIA2 (Triangular) plate structural elements were used.

---

5. Sciarra, J. J., VIBRATION REDUCTION BY USING BOTH THE FINITE ELEMENT STRAIN ENERGY DISTRIBUTION AND MOBILITY TECHNIQUES, 45th Shock and Vibration Symposium, Dayton, Ohio, August 1974.

6. Sciarra, J. J., USE OF THE FINITE ELEMENT DAMPED FORCED RESPONSE STRAIN ENERGY DISTRIBUTION FOR VIBRATION REDUCTION, U.S. Army Research Office - Durham, Final Report Contract DAH-CO4-71-C-0048, July 1974.

GEAR TOOTH MESH EXCITATION AND COMPLIANCE INPUT GENERATOR PROGRAM

PROGRAM DESCRIPTION

The original computer program for predicting spur gear tooth mesh excitation and compliance, which was entitled GEARO, was presented in Reference 2. GEARO was subsequently improved and renamed GGEAR (R-67) to include a sideband analysis capability and was modularized as indicated in Figure 3 to allow convenient inclusion of future improvements for helical and spiral bevel gears (Reference 7).

Under this contract, the analysis of the mesh dynamics for high-contact-ratio (HCR) spur gears was developed under a Boeing Vertol subcontract to Mechanical Technology Incorporated and was included as a subroutine in GGEAR. This revised program (GGEAR/HCR) has been completed and tested; the details of the HCR analysis and test cases are presented herein. The work accomplished is summarized below:

- Gear tooth mesh geometry analysis for HCR spur gears was developed.
- Tooth profiles, tooth bending, tooth contact surface deformation, and tooth support flexibility for meshes with two, three, or four pairs of teeth in contact instantaneously have been included.
- Load sharing between pairs of teeth, stiffness of individual tooth pairs and of the total mesh, and mesh excitation expressed in terms of deviation from true conjugate action tangent to the mesh pitch circle result from the analysis.
- Representative estimated values of tooth support flexibility are provided internally in the program with a user-controlled option for employing them.
- Restructured analytical expressions and input parameters to provide more efficient execution and improved utility were included.
- Automatic plotting capability for loads and tangential errors was completed.

---

7. Gu, A. L., and Badgley, R. H., PREDICTION OF GEAR-MESH-INDUCED HIGH-FREQUENCY VIBRATION SPECTRA IN GEARED POWER TRAINS, Mechanical Technology Incorporated, USAAMRDL Technical Report 74-5, U. S. Army Air Mobility Research and Development Laboratory, Fort Eustis, Virginia, January 1974, AD777496.

To apply the mesh excitation and compliance program to helical or bevel gears, it is currently necessary to employ the Tregold approximation (Reference 7). This method reduces a helical or bevel gear to an "equivalent" spur gear representation which is then input to GGEAR.

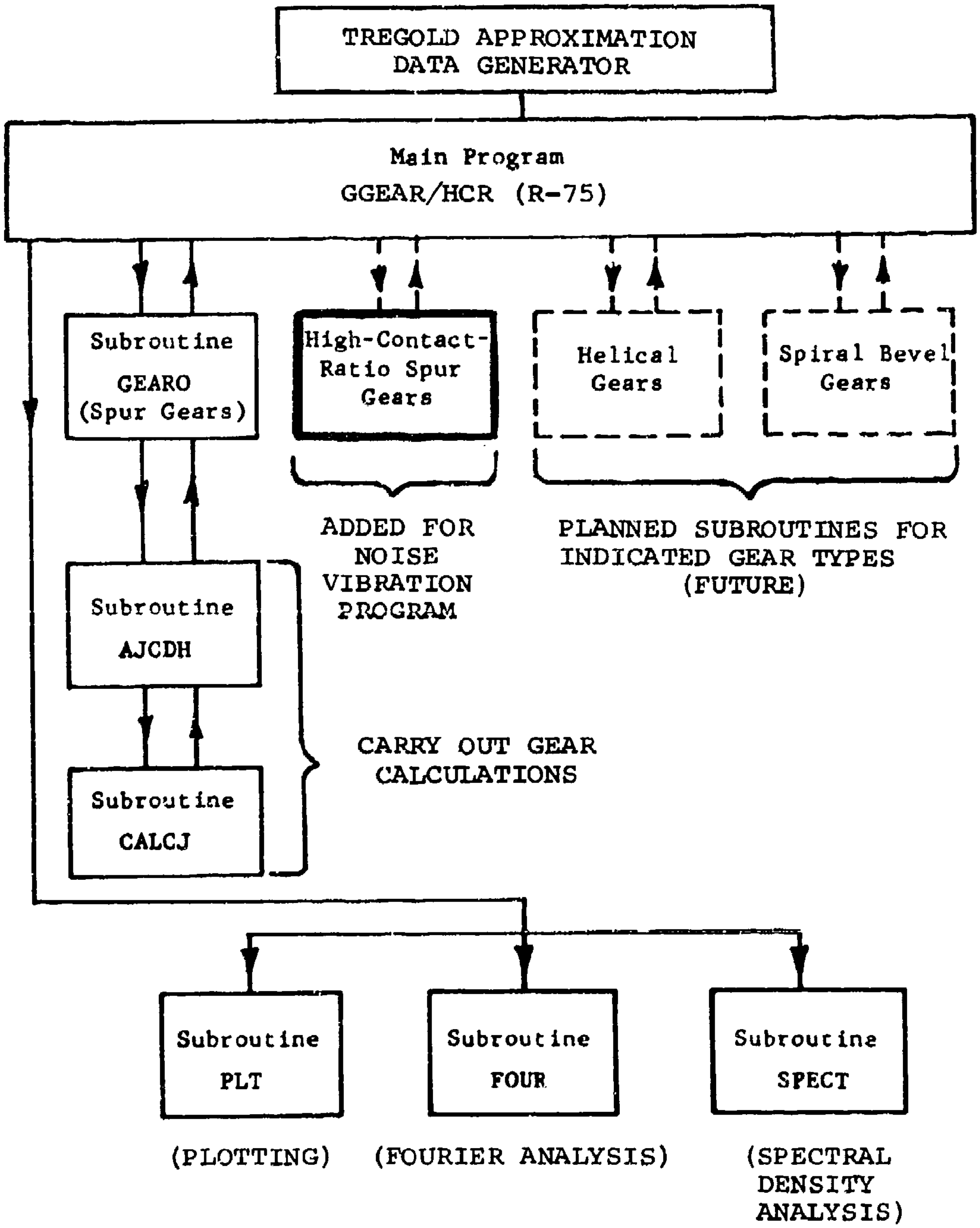


Figure 3. Gear Mesh Compliance/Excitation Computer Program Structure.

INPUT PREPARATION

A sample worksheet for developing input data required by Tregold's approximation for converting a spiral bevel gear into an equivalent spur gear is shown in Table 1. Table 2 shows data generated for the CH-47 forward rotor transmission spiral bevel input pinion and bevel ring gear. This data was used to fill in the data input sheet shown in Figure 4. A source listing of the Tregold's approximation program for calculating gear tooth mesh excitation and compliances is provided in Appendix A.

PROGRAM OUTPUT

The Tregold approximation program provides punched card output in a format for direct input into GGEAR (R-67). Also, it is convenient to use the Tregold program strictly as a data deck generator even when only spur gears are being considered. A sample output case is shown in Figure 5. In addition to the program calculated output data, the input data in Table 2 is also listed. This procedure provides a complete record of the computer run, including both input and output data. Also provided with the output listing is a set of punched cards for use as input into program GGEAR/HCR (R-75).

TABLE 1. TREGOLD'S APPROXIMATION-CONVERSION OF SPIRAL BEVEL INTO EQUIVALENT SPUR GEARS

		PINION			GEAR		
SYM	DEFINITION	SPIRAL	HELICAL	SPUR	SPIRAL	HELICAL	SPUR
N	NO. OF TEETH						
γ	PITCH CONE ANGLE						
	$\cos \gamma$						
	$\sin \gamma$						
ψ	SPIRAL ANGLE MEAN						
	$\cos \psi$						
	$\cos^2 \psi$						
	$\cos^3 \psi$						
N_v	$N/\cos \gamma$						
N_s	$N_v/\cos^3 \psi$						
F	FACE WIDTH						
f_v	F						
f_s	$F_v/\cos \psi$						
R							
R_M	$R-(F/2)\sin \gamma$						
R_v	$R_M/\cos \gamma$						
R_s	$R_v/\cos^2 \psi$						
α	R_M/R						
a	ADDENDUM @ LARGE END						
a_v	αa						
a_s	a_v						
R_{ov}	$R_v + a_v$						
R_{os}	$R_s + a_s$						
b	DEDENDUM @ LARGE END						
b_v	αb						
b_s	b_v						
R_{RV}	$R_v - b_v$						
R_{RS}	$R_s - b_s$						
d	WORKING DEPTH @ LARGE END						
d_M	αd						
d_v	d_M						
d_s	d_v						
TIF R_v	$R_{ov} - d_v$						
TIF R_s	$R_{os} - d_s$						
T	CIR. TOOTH THK. @ LARGE END						
T_M	$\alpha T \cos \psi$						
T_v	T_M						
T_s	T_v						
C_v	TIF $R_v - R_{RV}$						
C_s	TIF $R_s - R_{RS}$						
r_v	$.75\ C_v$						
r_s	$.75\ C_s$						

TABLE 2. CONVERSION OF SPIRAL BEVEL GEARS INTO EQUIVALENT SPUR GEARS FOR CH-47 FORWARD ROTOR TRANSMISSION

SYMBOL	DEFINITION	UNITS	PINION [114D1044(12)] SPIRAL BEVEL	PINION HELICAL	PINION SPUR	GEAR [114D1053(13)] SPIRAL BEVEL	GEAR HELICAL	GEAR SPUR
N	NUMBER OF TEETH OF SPIRAL BEVEL	—	29			51		
γ	PITCH CONE ANGLE OF SPIRAL BEVEL	DEGREES	31° 39'			67° 21'		
	COS γ	—	.85127			.38510		
	SIN γ	—	.52473			.92287		
ψ	SPIRAL ANGLE OF SPIRAL BEVEL - HELIX ANGLE OF EQUIVALENT HELICAL	DEGREES	25°	25°		25°	25°	
	COS ψ	—	.90631	.90631		.90631	.90631	
	$COS^2 \psi$	—	.82140	.82140		.82140	.82140	
	$COS^3 \psi$	—	.74444	.74444		.74444	.74444	
N_V	NUMBER OF TEETH OF EQUIVALENT HELICAL = $N/COS\ \gamma$	—		34.0655			132.4331	
N_S	NUMBER OF TEETH OF EQUIVALENT SPUR = $N_V/COS^3\ \psi$	—			45.7623			177.9058
F	FACE WIDTH OF SPIRAL BEVEL	INCHES	2.188			2.188		
F_V	FACE WIDTH OF EQUIVALENT HELICAL = F	INCHES		2.188			2.188	
F_S	FACE WIDTH OF EQUIVALENT SPUR = $F_V/COS\ \psi$	INCHES			2.4142			2.4142
R	PITCH RADIUS OF SPIRAL BEVEL AT LARGE END	INCHES	3.775			6.639		
R_M	MEAN PITCH RADIUS OF SPIRAL BEVEL = $R - (F/2)$ SIN γ	INCHES	3.2010			5.6294		
R_V	PITCH RADIUS OF EQUIVALENT HELICAL = $R_M/COS\ \gamma$	INCHES		3.7601			14.6180	
R_S	PITCH RADIUS OF EQUIVALENT SPUR = $R_V/COS^2\ \psi$	INCHES			4.5776			17.7964
α	RATIO OF RADII OF SPIRAL BEVEL = R_M/R	—	.8479			.8479		
a	ADDENDUM OF SPIRAL BEVEL AT LARGE END	INCHES	.297			.146		
a_V	ADDENDUM OF EQUIVALENT HELICAL = α a	INCHES		.2518			.1237	
a_S	ADDENDUM OF EQUIVALENT SPUR = a_V	INCHES			.2518			.1237
R_{OV}	OUTER RADIUS OF EQUIVALENT HELICAL = $R_V + a_V$	INCHES		4.0119			14.7417	
R_{OS}	OUTER RADIUS OF EQUIVALENT SPUR = $R_S + a_S$	INCHES			4.8294			17.9201
b	DEDENDUM OF SPIRAL BEVEL AT LARGE END	INCHES	.195			.346		
b_V	DEDENDUM OF EQUIVALENT HELICAL = α b	INCHES		.1653			.2933	
b_S	DEDENDUM OF EQUIVALENT SPUR = b_V	INCHES			.1653			.2933
R_{RV}	ROOT RADIUS OF EQUIVALENT HELICAL = $R_V - b_V$	INCHES		3.5948			14.3247	
R_{RS}	ROOT RADIUS OF EQUIVALENT SPUR = $R_S - b_S$	INCHES			4.4123			17.5031
d	WORKING DEPTH OF SPIRAL BEVEL AT LARGE END	INCHES	.443			.443		
d_M	MEAN WORKING DEPTH OF SPIRAL BEVEL = α d	INCHES	.3756			.3756		
d_V	WORKING DEPTH OF EQUIVALENT HELICAL = d_M	INCHES		.3756			.3756	
d_S	WORKING DEPTH OF EQUIVALENT SPUR = d_V	INCHES			.3756			.3756
$TIFR_V$	TRUE INVOLUTE FORM RADIUS OF EQUIVALENT HELICAL = $R_{OV} - d_V$	INCHES		3.6363			14.3661	
$TIFR_S$	TRUE INVOLUTE FORM RADIUS OF EQUIVALENT SPUR = $R_{OS} - d_S$	INCHES			4.4538			17.5445
T	CIRCULAR TOOTH THICKNESS OF SPIRAL BEVEL AT LARGE END	INCHES	.506			.312		
T_M	MEAN CIRCULAR TOOTH THICKNESS OF SPIRAL BEVEL = α T COS ψ	INCHES	.3888			.2397		
T_V	CIRCULAR TOOTH THICKNESS OF EQUIVALENT HELICAL = T_M			.3888			.2397	
T_S	CIRCULAR TOOTH THICKNESS OF EQUIVALENT SPUR = T_V	INCHES			.3888			.2397
C_V	RADIAL CLEARANCE OF EQUIVALENT HELICAL = $TIFR_V - R_{RV}$	INCHES		.0415			.0414	
C_S	RADIAL CLEARANCE OF EQUIVALENT SPUR = $TIFR_S - R_{RS}$	INCHES			.0415			.0414
r_V	TOOTH FILLET RADIUS OF EQUIVALENT HELICAL = $(.75)C_V$	INCHES		.0311			.0310	
r_S	TOOTH FILLET RADIUS OF EQUIVALENT SPUR = $(.75)C_S$	INCHES			.0311			.0310

TREGOLD'S APPROXIMATION

DATA INPUT SHEET Page 1 of __

GEAR TOOTH MESH EXCITATION AND COMPLIANCE PROGRAM

CARD	TITLE	SYMBOL	CASE 1	CASE 2	CASE 3	CASE 4
CARD 1	TITLE	EACH CASE MUST BE PRECEEDED BY A TITLE CARD. START TITLE IN CC2				
CARD 2	OUTPUT SELECTION 0 END EXECUTION 1 TREGOLD ONLY 3 R-67 (GGEAR) 2 R-33 (GEARO) 4 R-75 (GGEAR/HCR)	J				
	NO OF TEETH - DRIVING	TN1				
	NO OF TEETH - DRIVEN	TN2				
	PITCH CONE ANGLE (°) DRIVING	PCA1				
	PITCH CONE ANGLE (°) DRIVEN	PCA2				
	SPIRAL OR HELIX ANGLE (°) *	SA				
	FACE WIDTH (IN) - DRIVING	FW1				
	FACE WIDTH (IN) - DRIVEN	FW2				
	DIAMETRAL PITCH	PD				
	OUTSIDE BREAK RADIUS (IN) (+) / ADDENDUM (IN) (-) - DRIVING	RO1				
	OUTSIDE BREAK RADIUS (IN) (+) / ADDENDUM (IN) (-) - DRIVEN	RO2				
	ROOT RADIUS (IN) (+) / DEDENDUM (IN) (-) - DRIVING	RR1				
	ROOT RADIUS (IN) (+) / DEDENDUM (IN) (-) - DRIVEN	RR2				
	CIRCULAR TOOTH THICKNESS / PERCENT TO DRIVING (-) BACKLASH (+) - DRIVING	T1				
	CIRCULAR TOOTH THICKNESS / PERCENT TO DRIVING (-) BACKLASH (+) - DRIVEN	T2				
	PRESSURE ANGLE (°)	PA				
	CALCULATION POINTS	FI				
	TOOTH SPACE ERROR (IN) - DRIVING	TSE1				
	TOOTH SPACE ERROR (IN) - DRIVEN	TSE2				
	INTERNAL (0) OR EXTERNAL (1) **	MN				
	NO. OF FOURIER TERMS	MMM				
	TORQUE (IN-LB) - DRIVING	TORK				
	RPM - DRIVING	WS				
	PLOT (0 = NO, 1 = YES)	IPLT				
	DEVIATION FROM STANDARD CENTER DISTANCE (IN)	DEV				
	DEVIATION AND COMPLIANCE METHOD	NZJ				

NOTES: * For helical gears - input operating normal values.
** Program will not work if internal gear is driving.
Separate numbers by 2 spaces.

Figure 4. Data Input Sheet for Tregold's Approximation.

CARD

If NZJ = 0, 4, or 5

INSERT CARDS FOR TOOTH PROFILE DEVIATIONS (IN) –	DRIVING	UJ1
	DRIVEN	UJ2

(Format 6E13.5)

CARD

If NZJ = 0 or 3

INSERT CARDS FOR TOOTH COMPLIANCE (IN) –	DRIVING	ZJ1
	DRIVEN	ZJ2

(Format 6E13.5)

CARD

If TORK < 0

INSERT CARDS FOR VARIABLE TORQUE (IN-LB) –	DRIVING	TOR

(Format 6E13.5)

CARD

If NZJ = 1 or 4

INSERT CARDS FOR TOOTH PROFILE DEVIATION OF 4 POINTS ALONG PROFILE (OD, HP, LP, TIP) –	DRIVING
	DRIVEN

(Format 8E13.5)

CARD

If J < 0

MATERIAL PROPERTIES –	MODULUS OF ELASTICITY	– DRIVING	YE(1)
	POISSONS RATIO	– DRIVING	POS(1)
	MODULUS OF ELASTICITY	– DRIVEN	YE(2)
	POISSONS RATIO	– DRIVEN	POS(2)

Figure 4. Continued.

TREGOLD APPROXIMATION FOR EQUIVALENT SPUR GEARS

SPIRAL BEVEL GEAR DESIGN DATA GEAR NO. 1

NUMBER OF TEETH - SPIRAL BEVEL	29.000000	
PITCH CONE ANGLE - SPIRAL BEVEL	31.649970	DEGREES
SPIRAL ANGLE OF SPIRAL BEVEL	24.999980	DEGREES
FACE WIDTH OF SPIRAL BEVEL	2.188000	INCH
PITCH RADIUS OF SPIRAL BEVEL	3.775059	INCH
MEAN PITCH RADIUS OF SPIRAL BEVEL	3.201014	INCH
RATIO OF RADII OF SPIRAL BEVEL	0.847937	INCH
ADDENDUM OF SPIRAL BEVEL AT LARGE END	0.297000	INCH
DEDENDUM OF SPIRAL BEVEL AT LARGE END	0.195000	INCH
WORKING DEPTH OF SPIRAL BEVEL AT LARGE END	0.443000	INCH
CIR.TOOTH THICKNESS-SPIRAL BEVEL-LARGE END	0.506000	INCH
MEAN CIR.TOOTH THICKNESS-SPIRAL BEVEL	0.388858	INCH

EQUIVALENT HELICAL GEAR GEAR NO. 1

EQUIVALENT NUMBER OF TEETH - HELICAL	34.066570	
EQUIVALENT FACE WIDTH - HELICAL	2.188000	INCH
EQUIVALENT PITCH RADIUS - HELICAL	3.760261	INCH
EQUIVALENT ADDENDUM - HELICAL	0.251837	INCH
EQUIVALENT OUTER RADIUS - HELICAL	4.012097	INCH
EQUIVALENT DEDENDUM - HELICAL	0.165348	INCH
EQUIVALENT ROOT RADIUS - HELICAL	3.594913	INCH
EQUIVALENT WORKING DEPTH - HELICAL	0.375636	INCH
EQUIVALENT T.I.F. RADIUS - HELICAL	3.636460	INCH
EQUIVALENT CIRCULAR TOOTH THICKNESS-HELICAL	0.388858	INCH
EQUIVALENT RADIAL CLEARANCE-HELICAL	0.041548	INCH
EQUIVALENT TOOTH FILLET RADIUS-HELICAL	0.031161	INCH

EQUIVALENT SPUR GEAR GEAR NO. 1

EQUIVALENT NUMBER OF TEETH - SPUR	45.761130	
EQUIVALENT FACE WIDTH - SPUR	2.414182	INCH
EQUIVALENT PITCH RADIUS - SPUR	4.577871	INCH
EQUIVALENT ADDENDUM - SPUR	0.251837	INCH
EQUIVALENT OUTER RADIUS - SPUR	4.829708	INCH
EQUIVALENT DEDENDUM - SPUR	0.165348	INCH
EQUIVALENT ROOT RADIUS - SPUR	4.412523	INCH
EQUIVALENT WORKING DEPTH - SPUR	0.375636	INCH
EQUIVALENT T.I.F. RADIUS - SPUR	4.454071	INCH
EQUIVALENT CIRCULAR TOOTH THICKNESS-SPUR	0.388858	INCH
EQUIVALENT RADIAL CLEARANCE-SPUR	0.041548	INCH
EQUIVALENT TOOTH FILLET RADIUS-SPUR	0.031161	INCH

Figure 5. Sample Output from Tregold's Input Generator; Punched Output Provided for GGEAR/HCR (R-75).

TREGOLD APPROXIMATION FOR EQUIVALENT SPUR GEARS

SPIRAL BEVEL GEAR DESIGN DATA GEAR NO. 2

NUMBER OF TEETH - SPIRAL BEVEL	51.000000	
PITCH CONE ANGLE - SPIRAL BEVEL	67.349960	DEGREES
SPIRAL ANGLE OF SPIRAL BEVEL	24.999960	DEGREES
FACE WIDTH OF SPIRAL BEVEL	2.188000	INCH
PITCH RADIUS OF SPIRAL BEVEL	6.638896	INCH
MEAN PITCH RADIUS OF SPIRAL BEVEL	5.629280	INCH
RATIO OF RADII OF SPIRAL BEVEL	0.847924	INCH
ADDENDUM OF SPIRAL BEVEL AT LARGE END	0.146000	INCH
DEDENDUM OF SPIRAL BEVEL AT LARGE END	0.346000	INCH
WORKING DEPTH OF SPIRAL BEVEL AT LARGE END	0.443000	INCH
CIR.TOOTH THICKNESS-SPIRAL BEVEL-LARGE END	0.312000	INCH
MEAN CIR.TOOTH THICKNESS-SPIRAL BEVEL	0.239767	INCH

EQUIVALENT HELICAL GEAR GEAR NO. 2

EQUIVALENT NUMBER OF TEETH - HELICAL	132.426400	
EQUIVALENT FACE WIDTH - HELICAL	2.188000	INCH
EQUIVALENT PITCH RADIUS - HELICAL	14.616960	INCH
EQUIVALENT ADDENDUM - HELICAL	0.123797	INCH
EQUIVALENT OUTER RADIUS - HELICAL	14.740760	INCH
EQUIVALENT DEDENDUM - HELICAL	0.293382	INCH
EQUIVALENT ROOT RADIUS - HELICAL	14.323580	INCH
EQUIVALENT WORKING DEPTH - HELICAL	0.375630	INCH
EQUIVALENT T.I.F. RADIUS - HELICAL	14.365130	INCH
EQUIVALENT CIRCULAR TOOTH THICKNESS-HELICAL	0.239767	INCH
EQUIVALENT RADIAL CLEARANCE-HELICAL	0.041549	INCH
EQUIVALENT TOOTH FILLET RADIUS-HELICAL	0.031162	INCH

EQUIVALENT SPUR GEAR GEAR NO. 2

EQUIVALENT NUMBER OF TEETH - SPUR	177.886400	
EQUIVALENT FACE WIDTH - SPUR	2.414182	INCH
EQUIVALENT PITCH RADIUS - SPUR	17.795190	INCH
EQUIVALENT ADDENDUM - SPUR	0.123797	INCH
EQUIVALENT OUTER RADIUS - SPUR	17.918990	INCH
EQUIVALENT DEDENDUM - SPUR	0.293382	INCH
EQUIVALENT ROOT RADIUS - SPUR	17.501800	INCH
EQUIVALENT WORKING DEPTH - SPUR	0.375630	INCH
EQUIVALENT T.I.F. RADIUS - SPUR	17.543350	INCH
EQUIVALENT CIRCULAR TOOTH THICKNESS-SPUR	0.239767	INCH
EQUIVALENT RADIAL CLEARANCE-SPUR	0.041550	INCH
EQUIVALENT TOOTH FILLET RADIUS-SPUR	0.031162	INCH

Figure 5. Continued.

```
THE FOLLOWING IS THE PUNCHED INPUT FOR R-75 GGEAR/HCR


          CARD NUMBER 1

CH-47C FORWARD XMSN SPIRAL BEVEL MESH

          CARD NUMBER 2

         NMC          INT          MN          MMM          IPLT          IFOUR          ISPECT          IHICR
          1            0            1           10            0             1               1               0

          CARD NUMBER 2A

         IRM          IPERC          IRF          IPROP          ICENT          NZJ          IDIAG
          0              0            1             0              0              2             0

          CARD NUMBER 3

         FN1           FN2           DPIT          PANG           RO1          RO2 OR RI2
    45.761130    177.866400      4.998080      22.500000      4.829708      17.918990

          CARD NUMBER 4

         RM1          RF2          FI          T1
    4.412523    17.501800    12.000000    0.388858

          CARD NUMBER 5

          T2           F1            F2           RF1          RF2
    0.239767     2.414182     2.414182     0.031161     0.031162

          CARD NUMBER 7

          CL           VPT1          VPT2
    22.373000     0.000500     0.000500

          CARDS NUMBER 8C

         NTS
          1

         TQP
  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05
  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05
  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05
  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05  0.30400E 05
  0.30400E 05

          CARD NUMBER 9

         NWS          WS
          1          7460.0
```

Figure 5. Continued.

TREGOLD APPROXIMATION FOR EQUIVALENT SPUR GEARS

EQUIVALENT SPUR GEAR GEAR NO. 1

EQUIVALENT NUMBER OF TEETH - SPUR	28.000000	
EQUIVALENT FACE WIDTH - SPUR	1.770000	INCH
EQUIVALENT PITCH RADIUS - SPUR	2.799999	INCH
EQUIVALENT ADDENDUM - SPUR	0.200000	INCH
EQUIVALENT OUTER RADIUS - SPUR	2.999999	INCH
EQUIVALENT DEDENDUM - SPUR	0.232500	INCH
EQUIVALENT ROOT RADIUS - SPUR	2.567499	INCH
EQUIVALENT WORKING DEPTH - SPUR	0.399000	INCH
EQUIVALENT T.I.F. RADIUS - SPUR	2.600999	INCH
EQUIVALENT CIRCULAR TOOTH THICKNESS-SPUR	0.274500	INCH
EQUIVALENT RADIAL CLEARANCE-SPUR	0.033500	INCH
EQUIVALENT TOOTH FILLET RADIUS-SPUR	0.025125	INCH

Figure 5. Continued.

TREGOLD APPROXIMATION FOR EQUIVALENT SPUR GEARS

EQUIVALENT SPUR GEAR GEAR NO. 2

EQUIVALENT NUMBER OF TEETH - SPUR	39.000000	
EQUIVALENT FACE WIDTH - SPUR	1.550000	INCH
EQUIVALENT PITCH RADIUS - SPUR	3.900000	INCH
EQUIVALENT ADDENDUM - SPUR	0.199000	INCH
EQUIVALENT OUTER RADIUS - SPUR	4.098999	INCH
EQUIVALENT DEDENDUM - SPUR	0.217000	INCH
EQUIVALENT ROOT RADIUS - SPUR	3.683000	INCH
EQUIVALENT WORKING DEPTH - SPUR	0.399000	INCH
EQUIVALENT T.I.F. RADIUS - SPUR	3.699999	INCH
EQUIVALENT CIRCULAR TOOTH THICKNESS-SPUR	0.345800	INCH
EQUIVALENT RADIAL CLEARANCE-SPUR	0.016999	INCH
EQUIVALENT TOOTH FILLET RADIUS-SPUR	0.012749	INCH

Figure 5. Continued.

THE FOLLOWING IS THE PUNCHED INPUT FOR R-75 GGEAR/HCR

CARD NUMBER 1

CH47C FORWARD XMSN LOWER PLANETARY SUN/PLANET MESH

CARD NUMBER 2

NMC	INT	MN	MMM	IPLT	IFOUR	ISPECT	IHICR
1	0	1	10	0	1	1	0

CARD NUMBER 2A

IRM	IPERC	IRF	IPROP	ICENT	NZJ	IDIAG
0	0	1	0	0	2	0

CARD NUMBER 3

FN1	FN2	DPIT	PANG	RO1	RO2 OR RI2
28.000000	39.000000	5.000001	25.000030	2.999999	4.098999

CARD NUMBER 4

RP1	RP2	FT	T1
2.567490	3.683000	12.000000	0.274500

CARD NUMBER 5

T2	F1	F2	RF1	RF2
0.345800	1.770000	1.550000	0.025125	0.012749

CARD NUMBER 7

CL	VPT1	VPT2
6.699999	0.000300	0.000500

CARDS NUMBER 8C

NTS
1

TOP

0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05
0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05
0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05
0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05	0.13365E 05
0.13365E 05					

CARD NUMBER 9

NWS	WS
1	4242.0

Figure 5. Continued.

MESH DYNAMICS FOR HIGH-CONTACT-RATIO SPUR GEARS (R-75)

PROGRAM DESCRIPTION

A method has been developed for predicting the level and character of gear-mesh-induced vibrations in high-contact-ratio spur gears. The vibration amplitude is calculated with consideration of load sharing among pairs of engaging gear teeth. In addition to tooth compliance, which varies along the length of the tooth, tooth profile error and tooth spacing error are also included in the load sharing computation. The spectrum of the vibration is obtained by performing a Fourier analysis of the calculated mesh-induced profile of deviation from uniform rotation. The frequency spectra of gear-mesh-induced vibrations and the load sharing characteristics of a set of high-contact-ratio spur gears were calculated, and the results are discussed. This method of analysis is useful for designing low-vibration gears by properly controlling relevant gear parameters.

The meshing of gears introduces vibration because of the non-uniform deflection resulting from varying tooth compliance along the length of the tooth, errors in manufacturing, flexibility and accuracy of mountings, and varying tooth load through sharing during the course of meshing. This nonuniform deflection or displacement deviation introduces irregular motion, superimposed on the uniform rotation of gears, which becomes one of the major sources of vibration in a drive system. This irregularity of motion was recognized in the mid 1960's, and was analyzed as the initial step (Reference 2) in the development of an approach to the reduction of gear noise. Later efforts confirmed the precise source of the noise and recognized it as a mechanical vibration problem (References 1, 8, 9, 10, and 11). The validity of the entire concept and approach was experimentally verified as reported in Reference 12.

---

8. Badgley, R. H., MECHANICAL ASPECTS OF GEAR-INDUCED NOISE IN COMPLETE POWER TRAIN SYSTEMS, ASME Paper No. 70-WA/DGP-1, Presented at the ASME Winter Annual Meeting, New York, December 1970.

9. Badgley, R. H., GEARBOX DYNAMICS -- THE KEY TO UNDERSTANDING AND REDUCING ACOUSTIC-FREQUENCY ENERGY IN GEARED POWER TRAINS, Presented at the Meeting of the Aerospace Gearing Committee of the American Gear Manufacturers Association, Cleveland, Ohio, January 17-18, 1972.

10. Badgley, R. H., and Chiang, T., INVESTIGATION OF GEARBOX DESIGN MODIFICATIONS FOR REDUCING HELICOPTER GEARBOX NOISE, Mechanical Technology Incorporated, USAAMRDL Technical Report 72-6, U. S. Army Mobility Research and Development Laboratory, Fort Eustis, Virginia, March 1972, AD742735.

More recent efforts have incorporated this new approach into the design and redesign of gearbox components (References 4 and 13). Simultaneously, additional techniques have been added (References 7 and 14) to the mesh vibration analysis to treat various mesh irregularities. In this work a method of computing the gear-induced vibration and the associated frequency spectrum for high-contact-ratio spur gears is developed. The range of contact ratio under consideration is between 1 and 3; thus, there are up to four pairs of teeth in contact simultaneously for the spur gears being investigated. The method of load sharing calculation and the computation of mesh displacement deviation due to tooth bending, shear deflection, contact surface deformation, support flexibility, and tooth profile errors for high-contact-ratio spur gears are presented. These methods are generalized from those of Reference 7 which is limited to spur gears with contact ratio between one and two.

The limitations and assumptions in the calculations are:

1. Two cases of spur gears are treated: an external gear driving an external gear, and an external gear driving an internal gear.

2. The working portions of the tooth profiles are essentially involute. Design and manufacturing profile deviations are small enough not to affect load location, load direction, or tooth stiffness.

---

11. Badgley, R. H., REDUCTION OF NOISE AND ACOUSTIC-FREQUENCY VIBRATIONS IN AIRCRAFT TRANSMISSIONS, AHS Paper No. 661, Presented at the 28th Annual National Forum of the American Helicopter Society of Washington, D.C., May 1972.

12. Badgley, R. H., and Hartman, R. M., GEARBOX NOISE REDUCTION: PREDICTION AND MEASUREMENT OF MESH-FREQUENCY VIBRATIONS WITHIN AN OPERATING HELICOPTER ROTOR-DRIVE GEARBOX, ASME Paper No. 73-DET-31, Presented at the ASME Design Engineering Technical Conference, Cincinnati, Ohio, September 9-12, 1973.

13. Chiang, T., and Badgley, R. H., REDUCTION OF VIBRATION AND NOISE GENERATED BY PLANETARY RING GEARS IN HELICOPTER AIRCRAFT TRANSMISSIONS, ASME Paper No. 72-PTG-11, Presented at ASME Mechanisms Conference and International Symposium on Gearing and Transmissions, San Francisco, California, October 8-12, 1972.

14. Gu, A. L., and Badgley, R. H., PREDICTION OF VIBRATION SIDEBANDS IN GEAR MESHES, ASME Paper 74-DET-95, Presented at the Design Engineering Technical Conference, New York, New York, October 5-9, 1974.

3. There is no tip interference, either due to excessive addendum length or to tooth deflection under load.

4. In any single interval between the pitch points of two successive pairs of teeth, contact and load sharing are limited to a maximum of four consecutive pairs of teeth. In the same interval, there must at all times be at least one pair of teeth in contact and carrying load. Thus, the contact ratio of the gears is limited to between one and four.

5. The load is assumed to be transmitted uniformly across the face of the gear, except for normal end effects in stress distribution. This excludes any consideration of face crowning, lead modification, lead manufacturing error, gear windup, or nonuniform deflection of gear supports.

6. All variations in tooth deflection as the load point moves along the tooth profile are either confined to elastic effects on the tooth alone or are supplied as point-by-point compliances in the input data. This means that variations such as might result from the deflection of the thin rims are not calculated by the analysis.

7. Any lubricant film separating the mating teeth is assumed to have a constant thickness independent of the magnitude or the location of the transmitted load. The contact deformation is assumed to be independent of the lubricating film.

In the calculation of load sharing among tooth pairs in contact and the tooth deflection during a mesh cycle, tooth compliance is first computed point-by-point along the tooth profile for both the driving and driven gears. The basic kinematics of an involute spur gear mesh for high-contact-ratio spur gears is described below. Although external involute spur gears are used in the example, the case of an external gear driving an internal gear may be treated similarly. In Figure 6 , O is the gear center, $\varnothing_p$ the pressure angle, R_p the pitch radius, R_b the base radius, and R_o the outside radius. Subscript "1" stands for the driving gear and "2" for the driven gear. Segment b_1b_2 is the arc of action, which is the course of the contact point during mesh. Tooth contact begins at b_1, where the tip of the driven tooth first touches the flank of the driving tooth, and the mesh cycle ends at b_2, where the tip of the driving tooth last contacts the flank of the driven tooth. Figure 6 shows the contact occurring at the pitch point A. In Figure 7, X is an arbitrary contact point lying on the arc of action b_1b_2. V_1 and V_2 are the instantaneous velocities of the contact points on the two meshing teeth. The contact tangent plane is always perpendicular to the arc of action. P and P_t are the normal load and tangential load, respectively, shared by the pair of contacting teeth.

The base pitch p_b is the tooth-to-tooth distance between meshing pairs along the line of action. The mesh geometry analysis is limited to cases where

$$1 < m_p = \frac{\overline{b_1 b_2}}{p_b} < 3$$, and m_p is the contact ratio.

During one mesh cycle each of the two teeth in a meshing pair rotates an angle of $2\pi/N_i$, where N_i is the number of teeth for the driving gear (i = 1) or the driven gear (i = 2). For the purposes of computational convenience, the angle is divided into N_J equal intervals. These intervals form a scale which may be associated with the point of contact which moves along the involute profile as the gear rotates. Starting from a zero position when contact is at the pitch point, the successive values of J identify the intervals of rotation as

$$\alpha_{J1} = J_1 \frac{2\pi}{N_1 N_J} \tag{1}$$

$$\alpha_{J2} = J_2 \frac{2\pi}{N_2 N_J} \tag{2}$$

J is defined to be positive as contact moves outward on the involute and negative as it moves inward. Figure 8 shows a pitch-point contact of a pair of engaging teeth of two external involute spur gears, where R_M is the root circle radius and R_T defines the manufactured starting point of the involute profile. According to Equations (1) and (2), $\alpha_{J1} = \alpha_{J2} = 0$ for the tooth pair.

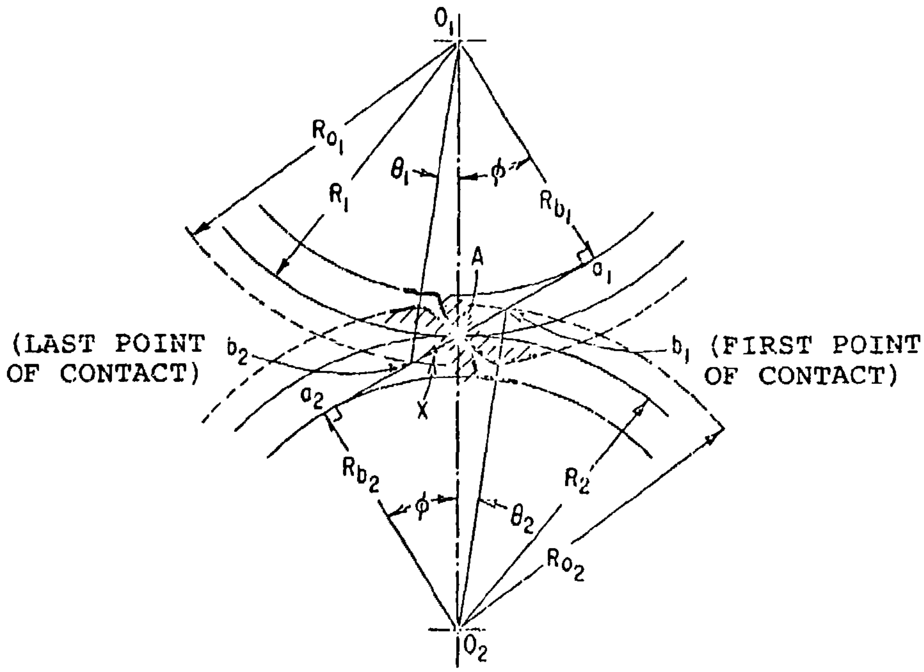


Figure 6. Involute Gears.

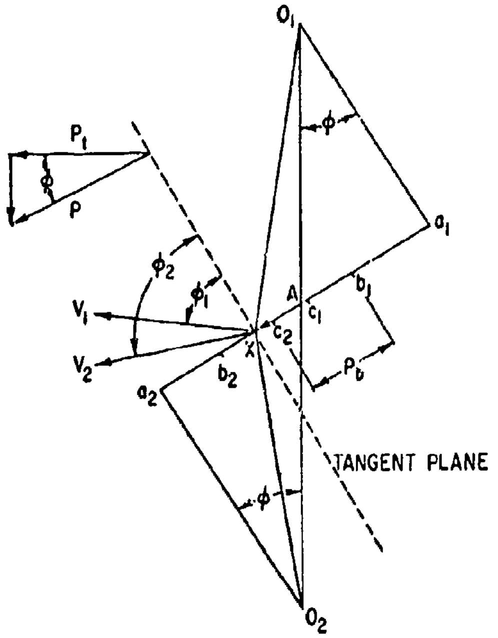


Figure 7. Tooth Mesh Kinematics.

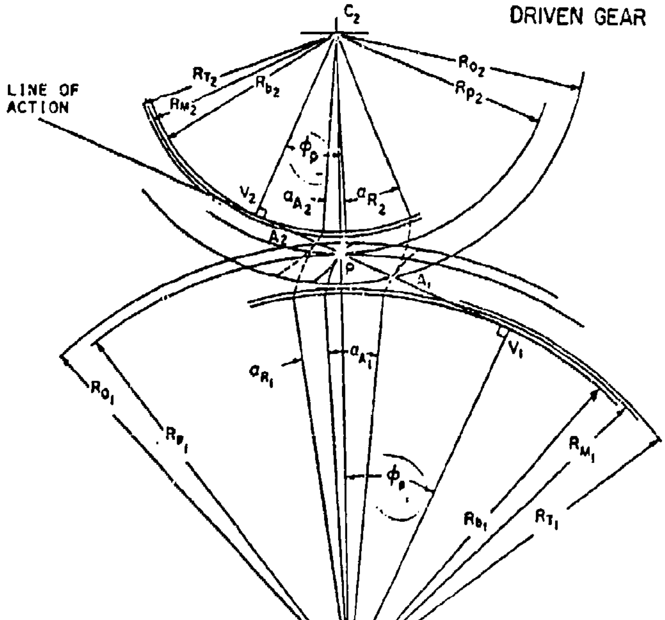


Figure 8. External Involute Spur Gears.

To check for engagement between the teeth of one meshing pair at any rotational position, the particular value of α_{J1} or α_{J2} is compared to the angles of approach and recess, α_{Ai} and α_{Ri} respectively, which limit the range of engagement (see Figure 8). Engagement is possible only when

$$\alpha_{R1} \geq \alpha_{J1} \geq \alpha_{A1} \tag{3}$$

and

$$\alpha_{R2} \geq \alpha_{J2} \geq \alpha_{A2} \tag{4}$$

It is noted that $\alpha_{A1} < 0$, since a rotation backward from the pitch-point contact is required. The angles of approach and recess may be expressed as follows (Reference 2):

$$\alpha_{A1} = -\frac{N_2}{N_1}\left[\sqrt{\rho_{o2}^{\,2} - 1} - \tan \phi_p\right] \tag{5}$$

$$\alpha_{R1} = \sqrt{\rho_{o1}^{\,2} - 1} - \tan \phi_p \tag{6}$$

$$\alpha_{A2} = -\frac{N_1}{N_2}\left[\sqrt{\rho_{o1}^{\,2} - 1} - \tan \phi_p\right] \tag{7}$$

$$\alpha_{R2} = \sqrt{\rho_{o2}^{\,2} - 1} - \tan \phi_p \tag{8}$$

where

$$\rho_{o1} = \frac{R_{o1}}{R_{b1}} \tag{9}$$

$$\rho_{o2} = \frac{R_{o2}}{R_{b2}} \tag{10}$$

When more than one pair of teeth are engaged in mesh instantaneously, the points of contact on the successive meshing teeth from the same gear have a fixed relationship as shown in Figure 9.

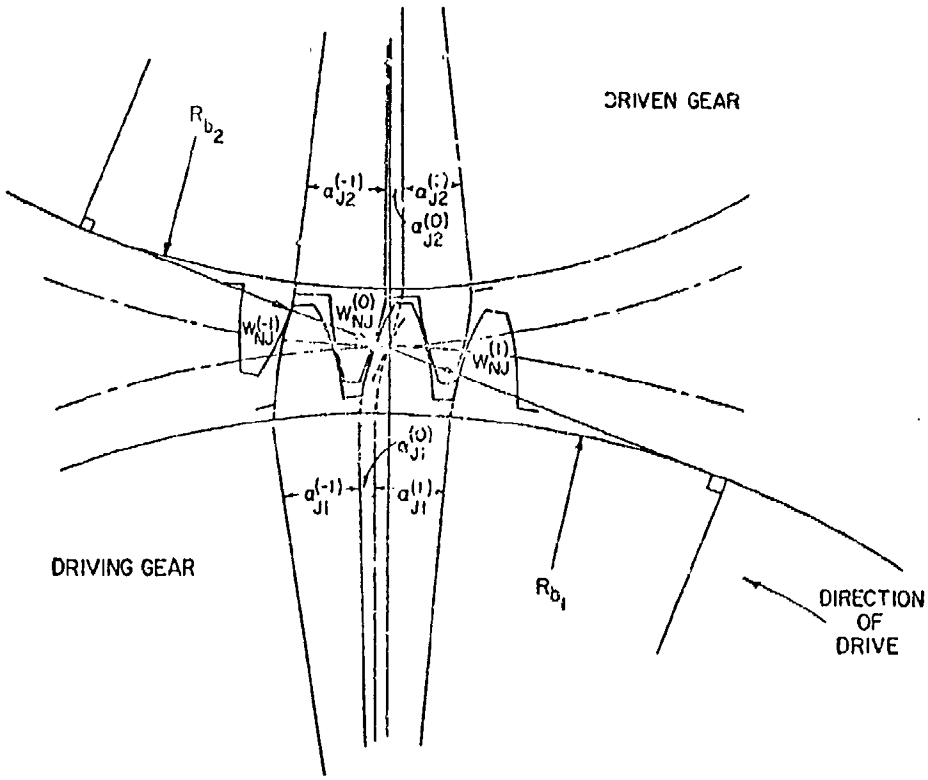


Figure 9. Three Successive Pairs of Teeth in Mesh.

$$\alpha_{J1}^{(-1)} + \alpha_{J1}^{(0)} = \alpha_{J1}^{(0)} + (-\alpha_{J1}^{(1)}) = \frac{2\pi}{N_1} \tag{11}$$

$$(-\alpha_{J2}^{(-1)}) + (-\alpha_{J2}^{(0)}) = (-\alpha_{J2}^{(0)}) + \alpha_{J2}^{(1)} = \frac{2\pi}{N_2} \tag{12}$$

where superscript (j) stands for jth pair. For high-contact-ratio spur gears in general, the above relationship may be generalized to

$$\left|\alpha_{J1}^{(j)}\right| + \left|\alpha_{J1}^{(j+1)}\right| = \frac{2\pi}{N_1} \tag{13}$$

$$\left|\alpha_{J2}^{(j)}\right| + \left|\alpha_{J2}^{(j+1)}\right| = \frac{2\pi}{N_2} \tag{14}$$

The division of transferred tooth load among the tooth pairs that are simultaneously in contact was determined from a consideration of gear tooth compliances and the relevant manufacturing errors for the contact points. Letting the meshing pairs be denoted by superscript (j) and (j) = (j_-), (j_-+1), ..., (-1), (0), (1) ..., (j_+ -1), and (j_+) successively, the total number of pairs of teeth in mesh instantaneously is ($|j_-|$ + j_+ + 1). It is noted that $j_- < 0$, and $j_+ > 0$. Using (0)th and (1)th pair of gears as representative, a schematic diagram of displacement deviation and load sharing relationships is shown in Figure 10. The gear teeth are represented as slender, spring-like members with deflections (B) caused by their respective loads. Profile errors (Z) at the points of contact are shown as buttons interacting with the deflections of the spring-teeth. Tooth spacing errors (V) are pictured as offsets in the projections of the bases of the spring-teeth. The deviation (V_{NT}) in the position of the driven gear relative to the driving gear of the (0)th pair, which results from the combination of deflections and errors, is shown by the offset between the base of the (0)th pair's spring-tooth on the driving gear and the base of the meshing spring-tooth on the driven gear. The profile errors, the tooth spacing errors, and the position deviations are all measured along the line of action (normal to the tooth surface). To separate the effects of profile error and tooth spacing, the former is defined as zero at the pitch point and the latter is measured between pitch points of successive teeth. To be consistent with the requirement of direction along the line of action, the tooth spacing error as measured between pitch points must be multiplied by the factor cos ϕ_p. (Z) is positive if material is added to the true involute, and

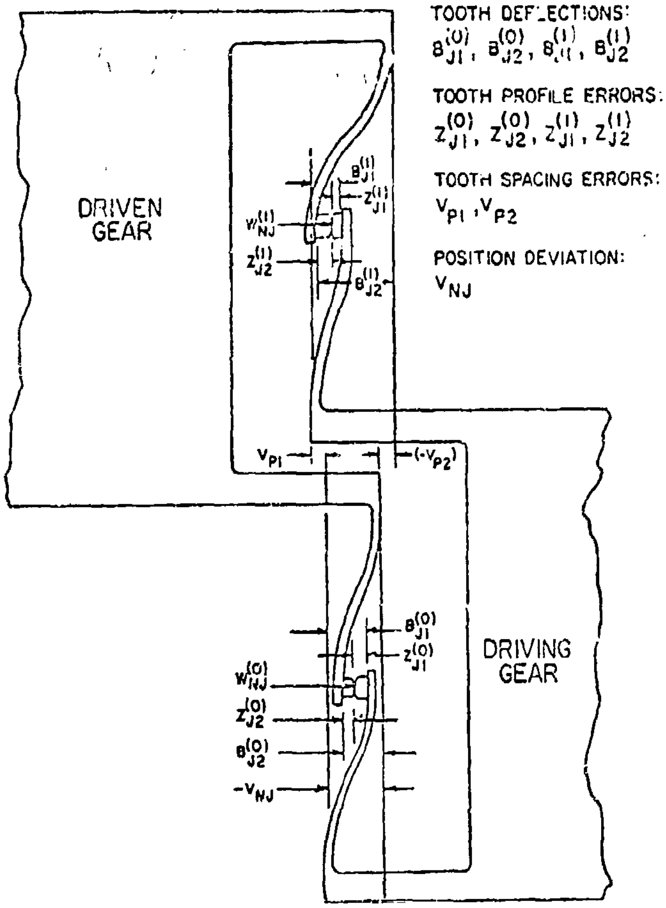


Figure 10. Schematic Diagram of Displacement Deviation and Load Sharing Relationships.

negative if material is subtracted; (V) is positive if the tooth spacing is smaller than the desired spacing.

The chain of displacements shown in Figure 10 yields the following equations:

$$-V_{NJ} = B_{J1}^{(0)} + B_{J2}^{(0)} - Z_{J1}^{(0)} - Z_{J2}^{(0)} \tag{15}$$

$$-V_{NJ} + V_{P1} - V_{P2} = B_{J1}^{(1)} + B_{J2}^{(1)} - Z_{J1}^{(1)} - Z_{J2}^{(1)} \tag{16}$$

For simplification, define

$$Z_J^{(j)} = Z_{J1}^{(j)} + Z_{J2}^{(j)} \tag{17}$$

and express deflections in terms of loads and elastic compliances of tooth as follows:

$$Q_J^{(j)} = Q_{J1}^{(j)} + Q_{J2}^{(j)} + U_{12}^{(j)} \tag{18}$$

where

$$B_{J1}^{(j)} + B_{J2}^{(j)} = W_{NJ}^{(j)} Q_J^{(j)} \tag{19}$$

In the above, $Q_{J1}^{(j)}$ and $Q_{J2}^{(j)}$ are, respectively, the tooth compliance at the Jth calculation point of the tooth profile of the driving and driven gears. $U_{J2}^{(j)}$ represents any additional combined tooth support compliance of the tooth pair.

Substituting Equations (17) and (18) into (15) and (16),

$$Q_J^{(0)} W_{NJ}^{(0)} + V_{NJ} = Z_J^{(0)} \tag{20}$$

$$Q_J^{(1)} W_{NJ}^{(1)} + V_{NJ} = Z_J^{(1)} + (V_{P1} - V_{P2}) \tag{21}$$

From the above derivation and the deflection and load sharing diagram shown in Figure 10, Equations (20) and (21) may be generalized to apply to the mesh system where there are ($|j_-|$ + j_+ + 1) pairs of teeth that contact instantaneously:

$$Q_J^{(j)} W_{NJ}^{(j)} + V_{NJ} = Z_J^{(j)} + (V_{P1} - V_{P2}) j \tag{22}$$

for $j_- \leq j \leq j_+$

The sum of the transferred loads is equal to the total normal tooth load,

$$\sum_{j=j_-}^{j_+} W_{NJ}^{(j)} = W_N \tag{23}$$

The system of Equations (22) and (23) consists of($|j_-|$ + j_+ + 2) linear equations in $W_{NJ}^{(j)}$ and V_{NJ}. It may be solved by standard method of matrix inversion to obtain the transferred load distribution $W_{NJ}^{(j)}$ and the normal displacement deviation V_{NJ}. If $W_{NJ}^{(j-)}$ is found to be less than zero, it means that the (j_-)th pair is no longer in contact. Consequently, $W_{NJ}^{(j-)}$ should be set equal to zero and a new j_- that equals the old j_- plus 1 should be used. Similarly, if $W_{NJ}^{(j+)}$ is found to be less than zero, $W_{NJ}^{(j+)}$ should be set zero and a new j_+ that equals the old j_+ minus 1 should be used.

The pitch line deviation V_J is related to the normal deviation V_{NJ} by

$$V_J = \frac{V_{NJ}}{\cos \phi_p} \tag{24}$$

V_J is computed and recorded from one calculation point to the next point by varying J from 0 to N_J. The total interval corresponds to one mesh cycle, and all teeth rotate an angle of $\frac{2\pi}{N_1}$. At $J = 0$, the (0)th pair may be set at the pitch-point position and thus $\alpha_{01}^{(0)} = \alpha_{02}^{(0)} = 0$. At any intermediate

J, $\alpha_{J1}^{(0)}$ and $\alpha_{J2}^{(0)}$ are calculated from Equations (1) and (2). The rotational positions of other engaging teeth may be obtained from Equations (13) and (14), and the engagement condition given in Equations (3) and (4) may be examined. At $J = N_J$ the (1)st pair of teeth moves to the pitch-point position and a new mesh cycle begins. The frequency spectrum of the gear-mesh-induced vibration is then obtained by a Fourier analysis of the tooth displacement deviations.

In the computation of mesh excitation expressed in terms of deviation from true conjugate action, tooth deflection due to elastic compliance as well as tooth profile error and tooth spacing error are considered. The consideration of the combined tooth compliance $Q_{12}^{(j)}$ is similar to that given in Reference 2. The combined compliance is the sum of the compliances of the two contacting teeth,

$$Q_{J12}^{(j)} = Q_{J1}^{(j)} + Q_{J2}^{(j)} \qquad (25)$$

where the subscript J stands for the Jth calculation point along the involute profile, and subscripts 1 and 2 represent the driving and driven gears respectively. Superscript (j) indicates the jth contacting pair of teeth.

The computation of $Q_{J1}^{(j)}$ and $Q_{J2}^{(j)}$ are similar; thus, only the compliance of the driving gear, $Q_{J1}^{(j)}$, will be considered.

The effective deflection at the contact point (load point) in the direction of the applied normal load is the quantity of interest. The total value of this deflection will be derived from the summation of four individual contributors, each formed by applying a simple deflection analysis. These four deflections are:

1. Displacement due to bending of the tooth as a cantilevered beam, $(B_{J1}^{(1)})_a$. The gear tooth is treated as a sequence of transverse segments, each of uniform rectangular cross-section. The depth of the uniform section is taken as the average of the depths at both ends of the segment. Figure 11 shows an external gear tooth with one such segment. Each segment will be identified by a K value

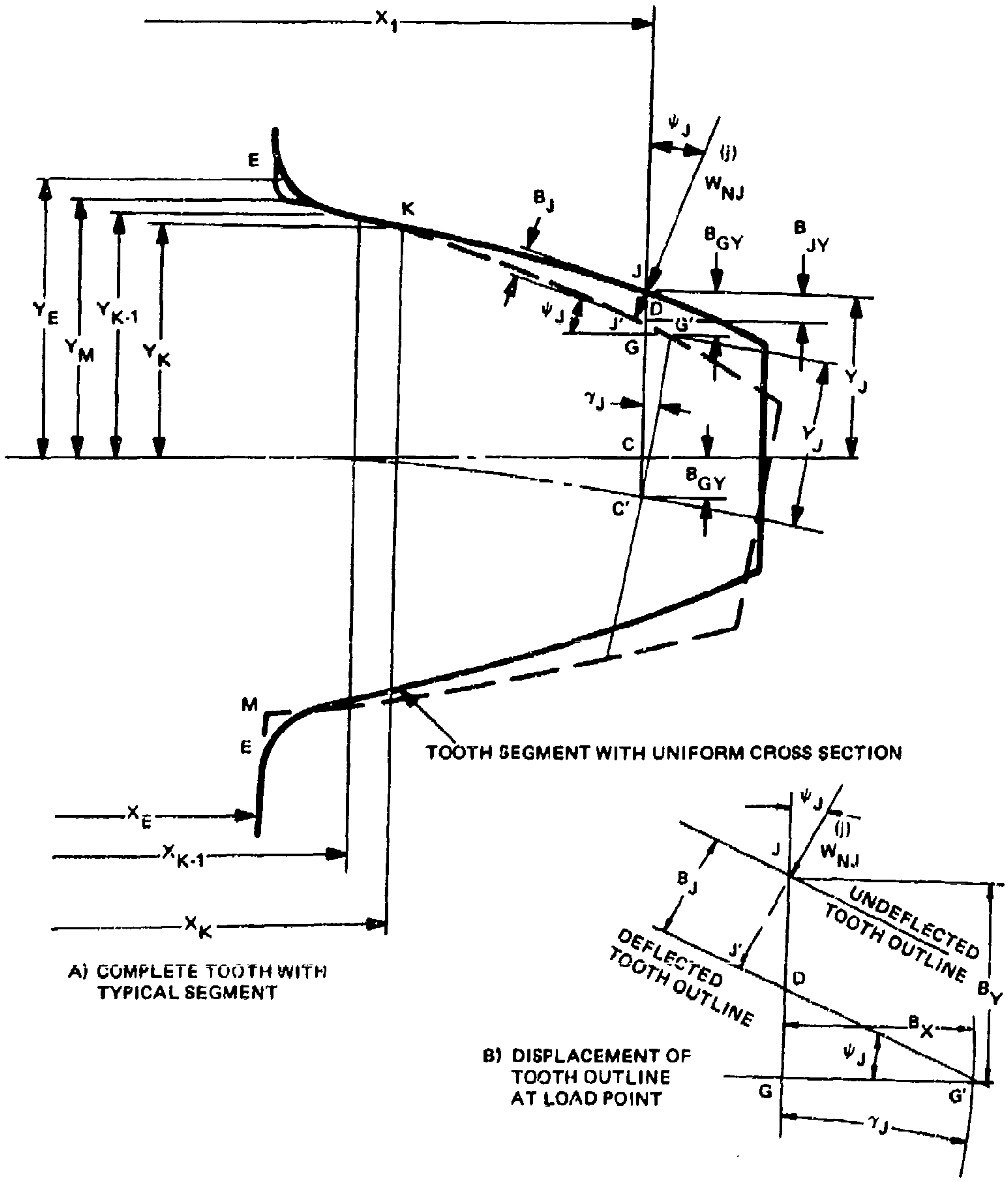


Figure 11. Beam Deflections in the Gear Tooth.

associated with its outer section, and the K value increases as the section moves out toward the tip of the tooth. The selection of the location of each K section may be made to correspond to the location of each of the N_J calculation points, as long as the N_J points are sufficiently close to each other. In Figure 11, J represents the point at which the load is applied. The angle ψ_J gives the direction of this load relative to the tooth centerline. The analysis which follows traces the contribution of each K segment to the displacement at the point J, which itself remains fixed. As the mating gear rotates, J moves into successive positions, and the deflection calculations must be repeated for each new position.

The moment of inertia of the rectangular cross-sectional area at K may be expressed in terms of the Y-coordinate and the tooth face width F.

$$I_K = \frac{1}{12} F (2Y_K)^3 = \frac{2FY_K^3}{3} \tag{26}$$

At the K-1 cross-section, which is the other end of the K segment,

$$I_{K-1} = \frac{2FY_{K-1}^3}{3} \tag{27}$$

For the assumed uniform cross-section of the segment between K and K-1, the mean of the two moments of inertia will be used.

$$\bar{I}_K = \frac{I_K + I_{K-1}}{2} \tag{28}$$

In the same manner, the mean area for the two cross-sections may be found:

$$A_K = F(2Y_K) = 2FY_K \tag{29}$$

$$A_{K-1} = 2FY_{K-1} \tag{30}$$

$$\bar{A}_K = \frac{A_K + A_{K-1}}{2} \tag{31}$$

The length of the K segment is

$$L_K = X_K - X_{K-1} \tag{32}$$

The distance of the K segment, at its outer end, from the point J of load application is similarly found:

$$S_{JK} = X_J - X_K \tag{33}$$

In Figure 11, the normal load at J acting on the cantilevered gear tooth is equivalent to a concentrated transverse load equal to $W_{NJ}^{(j)} \cos \psi_J$, and a counterclockwise moment being $W_{NJ}^{(j)} \sin \psi_J \, Y_J$.

In the bending of the tooth, each of these load conditions causes a transverse displacement and a rotation of the tooth centerline under the point of load application. These two displacements will be evaluated for each of the two load conditions and then combined.

The first case considered is that of a tooth segment subjected to a transverse load applied some distance away, with the rest of the gear tooth serving as a rigid support of a rigid extension (Figure 12a).

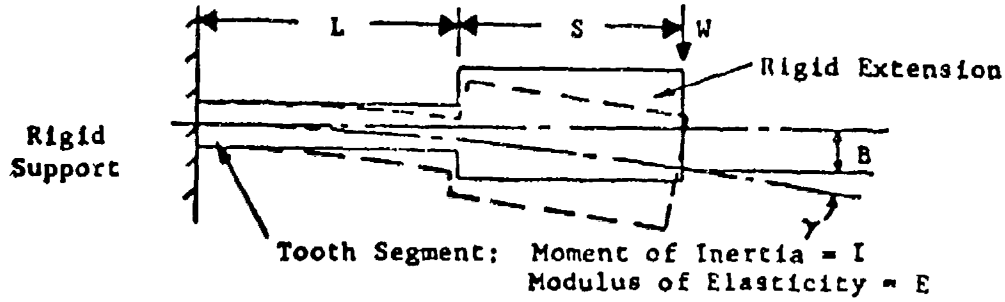
From standard beam deflection theory, the transverse deflection at the load is

$$\left|B_{Y1K}^{(j)}\right|_{a1} = \frac{W_{NJ}^{(j)} \cos \psi_J \, L_K}{2\, E_\nu \, \bar{I}_K} \left(L_K^2 + 3S_{JK}L_K + 3S_{JK}^{\;2}\right) \tag{34}$$

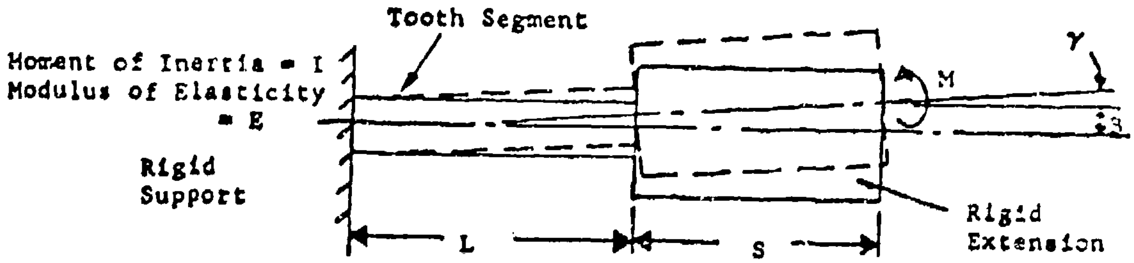
The centerline rotation at the load is

$$\left|\gamma_{J1K}^{(j)}\right|_{a1} = \frac{W_{NJ}^{(j)} \cos \psi_J \, L_K}{2\, E_\nu \, \bar{I}_K} \left(L_K + 2S_{JK}\right) \tag{35}$$

In the above, E_ν is the effective bending modulus of elasticity for gears whose face width is large compared to the tooth thickness:



a. Under a Concentrated Load.



b. Under a Moment.

Figure 12. Beam Deflection Diagrams.

$$E_\nu = \frac{E}{1 - \nu^2} \tag{36}$$

where E is the Young's modulus of elasticity and ν the Poisson's ratio. If the gear tooth is narrow relative to its thickness, the factor $\frac{1}{1 - \nu^2}$ should be eliminated.

The second case considered is that of the tooth segment subjected to a moment load applied some distance away, with the rest of the gear tooth serving as a rigid support or a rigid extension (Figure 12b).

From the beam deflection theory, the transverse deflection at the load is

$$\left(B_{Y1K}^{(j)}\right)_{a2} = \frac{-W_{NJ}^{(j)} \sin \psi_J \, Y_J L_K}{2 \, E_\nu \, \bar{I}_K} (L_K + 2S_{JK}) \tag{37}$$

The centerline rotation at the load is

$$\left(\gamma_{J1K}^{(j)}\right)_{a2} = \frac{-W_{NJ}^{(j)} \sin \psi_J \, Y_J \, L_K}{E_\nu \, \bar{I}_K} \tag{38}$$

The deflections for each tooth segment may be summed to yield the total transverse deflection at the load $(B_{Y1}^{(j)})_{a1}$ and $(B_{Y1}^{(j)})_{a2}$, and the centerline rotation at the load $(\gamma_{J1}^{(j)})_{a1}$ and $(\gamma_{J1}^{(j)})_{a2}$.

The displacement in the direction of the load (Figure 12) due to a transverse deflection B_Y and a centerline rotation γ_J is

$$B_J = B_Y \cos \psi_J - Y_J \gamma_J \sin \psi_J \tag{39}$$

By summing the deflections for all tooth segments and computing the displacement due to the concentrated load and moment load together, the total displacement of the tooth in the load direction is

$$\left(B_{J1}^{(j)}\right)_a = \frac{W_{NJ}^{(j)}}{E_\nu \bar{I}_K}\left[\frac{\cos^2\psi_J}{3}\sum L_K(L_K^2 + 3S_{JK}L_K + 3S_{JK}^2)\right.$$

$$- \cos\psi_J \sin\psi_J \cdot Y_J \sum L_K(L_K + 2S_{JK})$$

$$\left. + \sin^2\psi_J \cdot Y_J^2 \sum L_K\right] \tag{40}$$

which gives the total deflection from the bending of the tooth loaded as a cantilevered beam.

2. Displacement due to the shear deformation of the tooth as a cantilevered beam, $(B_{J1}^{(j)})_b$. The shear deflection of the tooth as a cantilevered beam is also caused by the transverse component of the applied load, $W_{NJ}^{(j)} \cos \Psi_J$. This deflection offsets the centerline without rotating any transverse section. The deflection at the load point then becomes the sum of the deflections for each of the tooth segments defined previously.

   The individual segment subjected to the shear load is represented in Figure 13, with the rest of the gear tooth appearing as a rigid support or a rigid extension.

   From standard methods for beam shear deflection analysis, the transverse deflection when the cross-section is rectangular is

$$\left(B_{Y1K}^{(j)}\right)_b = \frac{1.2W_{NJ}^{(j)} \cos\psi_J \cdot L_K}{G\bar{A}_K} \tag{41}$$

$$(Y_{J1K}^{(j)})_b = 0 \tag{42}$$

   where G is the shear modulus of elasticity.

   Summing the deflection for all segments from the base of the tooth to the applied load point J,

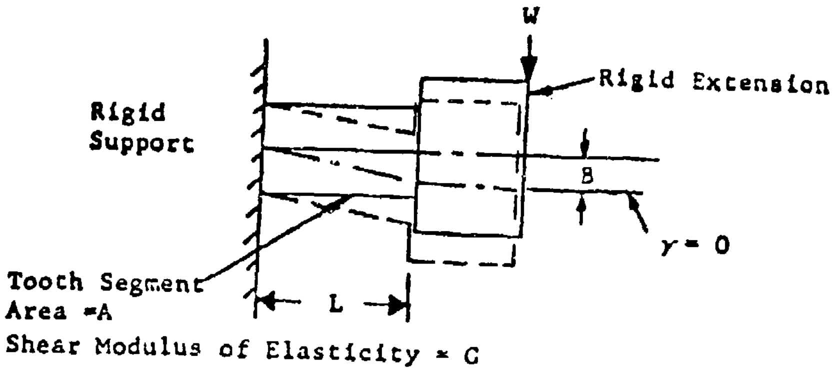


Figure 13. Shear Deflection Diagram.

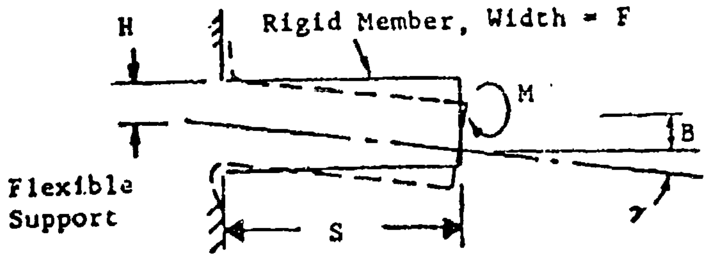


Figure 14. Base Rotation Deflection Diagram.

$$\left(B_{Y1}^{(j)}\right)_b = \frac{1.2W_{NJ}^{(j)} \cos \psi_J}{G} \sum \frac{L_K}{\bar{A}_K} \tag{43}$$

$$\left(Y_{J1}^{(j)}\right)_b = 0 \tag{44}$$

To obtain the resulting displacement due to the shear deformation of the tooth as a cantilevered beam in the direction of the load, the factor cos ψ_J is multiplied by $(B_{Y1}^{(j)})_b$.

$$\left(B_{J1}^{(j)}\right)_b = \frac{1.2W_{NJ}^{(j)} \cos^2 \psi_J}{G} \sum \frac{L_K}{\bar{A}_K} \tag{45}$$

3. Displacement due to the rotation of the tooth in the supporting structure at its base, $(B_{J1}^{(j)})_c$. The moment that causes this rotation is due to the applied load. The transverse component of this load $W_{NJ}^{(j)}$ cos ψ_J will create the moment $W_{NJ}^{(j)}$ cos $\psi_J \cdot S_J$, where S_J is the distance from the load point to the base of the tooth. The axial component, as noted previously, will create the moment $W_{NJ}^{(j)}$ sin ψ_J.

   Figure 14 shows this loading and the resulting deflection in simplified form.

   The equation for the displacements at the load point from Reference 15 is

$$\left(B_{Y1}^{(j)}\right)_c = \frac{1.327W_{NJ}^{(j)} (\cos \psi_J \cdot S_J - \sin \psi_J \cdot Y_J)S_J}{E_\nu Y_E^2 F} \tag{46}$$

$$\left(Y_{J1}^{(j)}\right)_c = \frac{1.327W_{NJ}^{(j)} (\cos \psi_J \cdot S_J - \sin \psi_J \cdot Y_J)}{E_\nu Y_E^2 F} \tag{47}$$

---

15. O'Donnel, W. J., STRESS AND DEFLECTION IN BUILT-IN BEAMS, ASME Paper No. 62-WA-16, 1962.

The resulting displacement in the direction of the load due to the rotation of the tooth in its supporting structure is

$$\left|B_{J1}^{(j)}\right|_c = \frac{1.327W_{NJ}^{(j)}\,(\cos\psi_J \cdot S_J - \sin\psi_J \cdot Y_J)(\cos\psi_J \cdot S_J - \sin\psi_J \cdot Y_J)}{E_\nu\, Y_E^2\, F}$$

$$= \frac{1.327W_{NJ}^{(j)}(\cos\,\psi_J \cdot S_J - \sin\,\psi_J \cdot Y_J)^2}{E_\nu Y_E^2 F} \qquad (48)$$

4. Displacement due to the contact or Hertzian deformation, $B_{J12}^{(j)})_d$. This deformation is based on a semiempirical equation (Reference 16) developed for a contacting cylinder in roller bearings,

$$B_{12} = \frac{3\times10^{-4}\,\epsilon_E^{2.7}\,W^{0.9}}{F_{12}^{0.8}} \qquad (49)$$

where

$$\epsilon_E = \sqrt[3]{\frac{11{,}500(E_{\nu1} + E_{\nu2})}{(E_{\nu1}\,E_{\nu2})}} \qquad (50)$$

This equation treats the general case where the materials of the two mating surfaces may have different properties. Subscripts 1 and 2 are used to refer to the two surfaces. The deflection given by the equation is for the two contracting surfaces combined; hence, the notation 12 is used in the subscript. Since the face widths of the two surfaces may be unequal, F_{12} refers to the effective face width of the combination. A suitable approximation of this effective face width is the smaller of the two widths.

After the substitution of the equation for a composite modulus of elasticity,

$$\frac{1}{E_{\nu12}} = \frac{1}{2}\left(\frac{1}{E_{\nu1}} + \frac{1}{E_{\nu2}}\right) = \frac{(E_{\nu1} + E_{\nu2})}{2(E_{\nu1}E_{\nu2})} \qquad (51)$$

16. Palmgren, David, BALL AND ROLLER BEARING ENGINEERING, 3rd Edition, SKF Industries, Inc.

into Equation (50), the result is combined with Equation (49) and this becomes

$$B_{12} \doteq \frac{2.55\, W^{0.9}}{E_{\nu 12}^{0.9}\, F_{12}^{0.8}} \qquad (52)$$

There is a major difference between the conditions applying to a roller between bearing races and those applying to a loaded gear tooth. Although in both cases contact is between cylindrical surfaces, the nature of the support of the bodies with these cylindrical surfaces is quite different. The force at one contacting surface on the roller is opposed by an equal force located diametrally opposite. As a result the entire roller is loaded with the full compressive force, and the deflection given by Equation (52) is the deflection at the contact point and throughout the roller. A gear tooth does not provide the full cylindrical cross-section of the roller. Furthermore, the applied load is not supported by a diametrally opposed force but by the shear forces distributed across the base of the tooth. The first item calls attention to a reduced material cross-section available for deformation. The second item points out that, due to the side support, the internal loading in the cross-section of the tooth is not constant but reduces with increased distance from the contact point. These differences make the contact deflection in gear teeth significantly less than indicated by Equation (52), and the estimate used for this reduction in deflection is 50 percent. This correction gives calculated results which are still somewhat larger than the experimental results to which they were compared. However, the apparent excess in calculated contact deflection will be more than offset in many practical applications by the increase in deflection resulting from slight errors in the contacting surfaces.

By including the 50 percent correction, Equation (52) may be rewritten to adapt more closely to the displacement due to Hertzian deformation of contacting gear teeth.

$$\left(B_{J12}^{(j)}\right)_d = \frac{1.275\, W_{NJ}^{(j)\,0.9}}{E_{\nu 12}^{0.9}\, F_{12}^{0.8}} \qquad (53)$$

From the four displacement equations, (40), (45), (48), and (53), the compliance may be computed from

$$Q_{J1}^{(j)} = \frac{B_{J1}^{(j)}}{W_{NJ}^{(j)}} \tag{54}$$

From Equations (40), (45), and (48),

$$\left|Q_{J1}^{(j)}\right|_a = \frac{1}{E_\nu I_K}\left[\frac{\cos^2 \psi_J}{3}\sum L_K(L_K^2 + 3S_K L_K + 3S_{JK}^2) - \cos\psi_J \sin\psi_J \cdot Y_J \sum L_K(L_K + 2S_{JK}) + \sin^2\psi_J \cdot Y_J^2 \sum LK\right] \tag{55}$$

$$\left|Q_{J1}^{(j)}\right|_b = \frac{1.2\cos^2\psi_J}{G}\sum \frac{L_K}{\bar{A}_K} \tag{56}$$

$$\left|Q_{J1}^{(j)}\right|_a = \frac{1.327(\cos\psi_J \cdot S_J - \sin\psi_J \cdot Y_J)^2}{E_\nu Y_E^2 F} \tag{57}$$

These three compliances combined may be viewed as the total beam compliance of the driving gear tooth of the jth pair

$$\left|Q_{J1}^{(j)}\right|_{abc} = \left|Q_{J1}^{(j)}\right|_a + \left|Q_{J1}^{(j)}\right|_b + \left|Q_{J1}^{(j)}\right|_c \tag{58}$$

The total beam compliance of the driven gear, $Q_{J2}^{(j)}$, may be similarly computed.

The contact deformation compliance may be obtained by combining Equations (53) and (54):

$$\left|Q_{J12}^{(j)}\right|_d = \frac{1.275}{E_{\nu 12}^{0.9} F_{12}^{0.8} \left|W_{NJ}^{(j)}\right|^{0.1}} \tag{59}$$

Equation (59) shows that the compliance is somewhat dependent on the value of the applied load. Since the load appears to the one-tenth power, there will be little variation in compliance over a reasonably wide range of load, and $W_{NJ}^{(j)}$ may be replaced by an average value covering all conditions of loading without appreciable error. A suitable average value would be one-half the total load carried by the gear teeth,

$$W_{NJ}^{(j)} = .5\, W_N \tag{60}$$

and then Equation (59) becomes

$$\left|Q_{J12}^{(j)}\right|_d = \frac{1.37}{E_{v12}^{0.9}\, F_{12}^{0.8}\, W_N^{0.1}} \tag{61}$$

Because this relationship is independent of surface curvature, it will apply for all load positions and for both the case of two external gears and the case of one external and one internal gear. The contact compliance requires the additional input information of the total transmitted load W_N and, in this respect, differs from the other compliance components.

In summary, the total elastic compliance of the jth pair mating gear teeth may be found by applying Equations (55), (56), and (57) for each of the gears, combining the results, and then adding the contact compliance for the two gears in combination.

$$Q_{J12}^{(j)} = \left|Q_{J1}^{(j)}\right|_{abc} + \left|Q_{J2}^{(j)}\right|_{abc} + \left|Q_{J12}^{(j)}\right|_d \tag{62}$$

SAMPLE GEAR MESH DYNAMICS CALCULATIONS

A set of high-contact-ratio spur gears, described in Table 3, was studied using the analysis previously described. The meshing error, tangential tooth load sharing, and frequency spectrum of mesh induced vibration are shown in Figures 15, 16, and 17 respectively. In each figure, there are three curves representing gears with (a) perfect tooth profile without spacing error, (b) the tooth profile modifications listed in Table 4, but with no spacing error, and (c) tooth modification and spacing errors of 0.000187 in. for the driving gear and -0.000187 in. for the driven gear.

The tooth meshing error, or deviation from pure conjugate action, as a pitch-line linear measurement of the motion of the driving gear leading the driven gear is shown in Figure 15 as the solid curve. The mesh cycle is represented by 25 calculation points. In the initial period of the mesh cycle there are three pairs of teeth in contact simultaneously. The beginning of the mesh cycle corresponds to the pitch-point contact of the middle pair. In the middle of the cycle there are only two pairs in contact; therefore, the tooth meshing error experiences a step change. Near the end of the cycle a new pair of teeth engages in the mesh and the meshing error drops accordingly. The end of the mesh cycle is again a pitch-point contact.

The tangential load shared by a tooth during its meshing engagement is represented by the solid curve given in Figure 16. A step change of the tangential load corresponds to the point at which another pair of teeth comes in or goes out of engagement. The pitch-point contact is identified in Figure 16. The mesh cycle over which the tooth meshing error is presented in Figure 15 is indicated. The cycle begins when the contact is at the pitch-point of the tooth, and ends when the following tooth gets to the pitch-point contact position.

The frequency spectrum of the gear-mesh-induced vibration is shown in Figure 17 and is identified by the case of $Z_J = V_P = 0$. In the figure, f_M is the mesh frequency. It is seen that the spectrum is discrete and the amplitude decreases with increasing frequency.

TABLE 3. GEAR DATA

	DRIVING GEAR	DRIVEN GEAR
Number of Teeth	33	58
Diametral Pitch, 1/in.	4.55	4.55
Pressure Angle, deg	21	21
Outer Diameter, in.	7.900	7.900
Root Diameter, in.	6.594	6.038
Circular Tooth Thickness at Pitch Diameter, in.	0.369	0.314
Tooth Face Width, in.	0.5	0.5
Tooth Fillet Radius, in.	0.087	0.083
Center Distance, in.	10	10
Young's Modulus, lb/in.^{2}	3×10^7	3×10^7
Shear Modulus, lb/in.^{2}	1.15×10^7	1.15×10^7
Poisson's Ratio	0.3	0.3
Driving Torque, in.-lb	9831	

TABLE 4. TOOTH PROFILE MODIFICATION

Driving Gear

Z_J (in.)	-0.00095	-0.00008	0	-0.00009	-0.00075
Roll Angle (deg)	34.43	28	21.99	18	9.57

Driven Gear

Z_J (in.)	-0.00075	-0.0001	0	-0.0003	-0.00055
Roll Angle (deg)	29.06	24.56	21.99	16.65	14.91

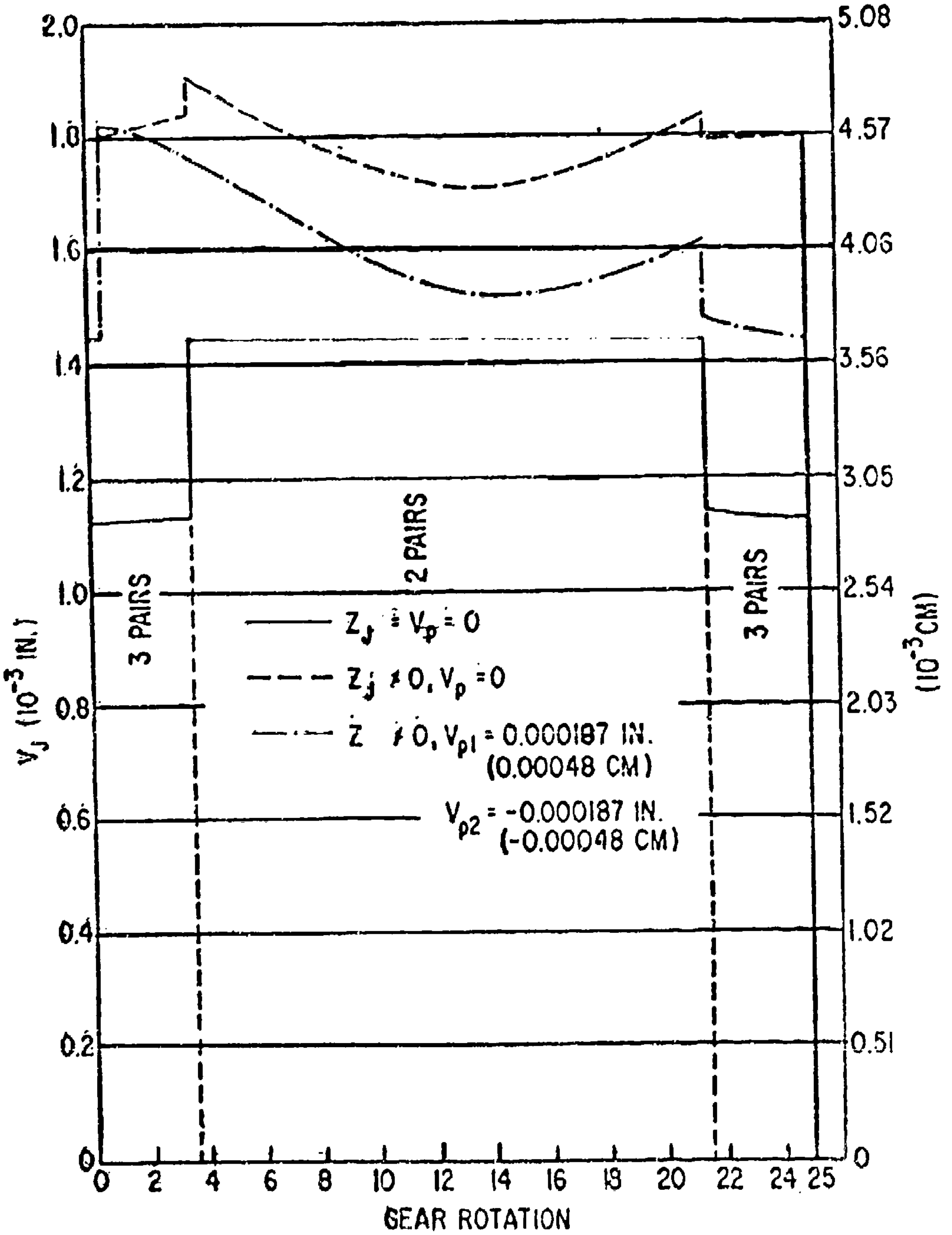


Figure 15. Tooth Meshing Error.

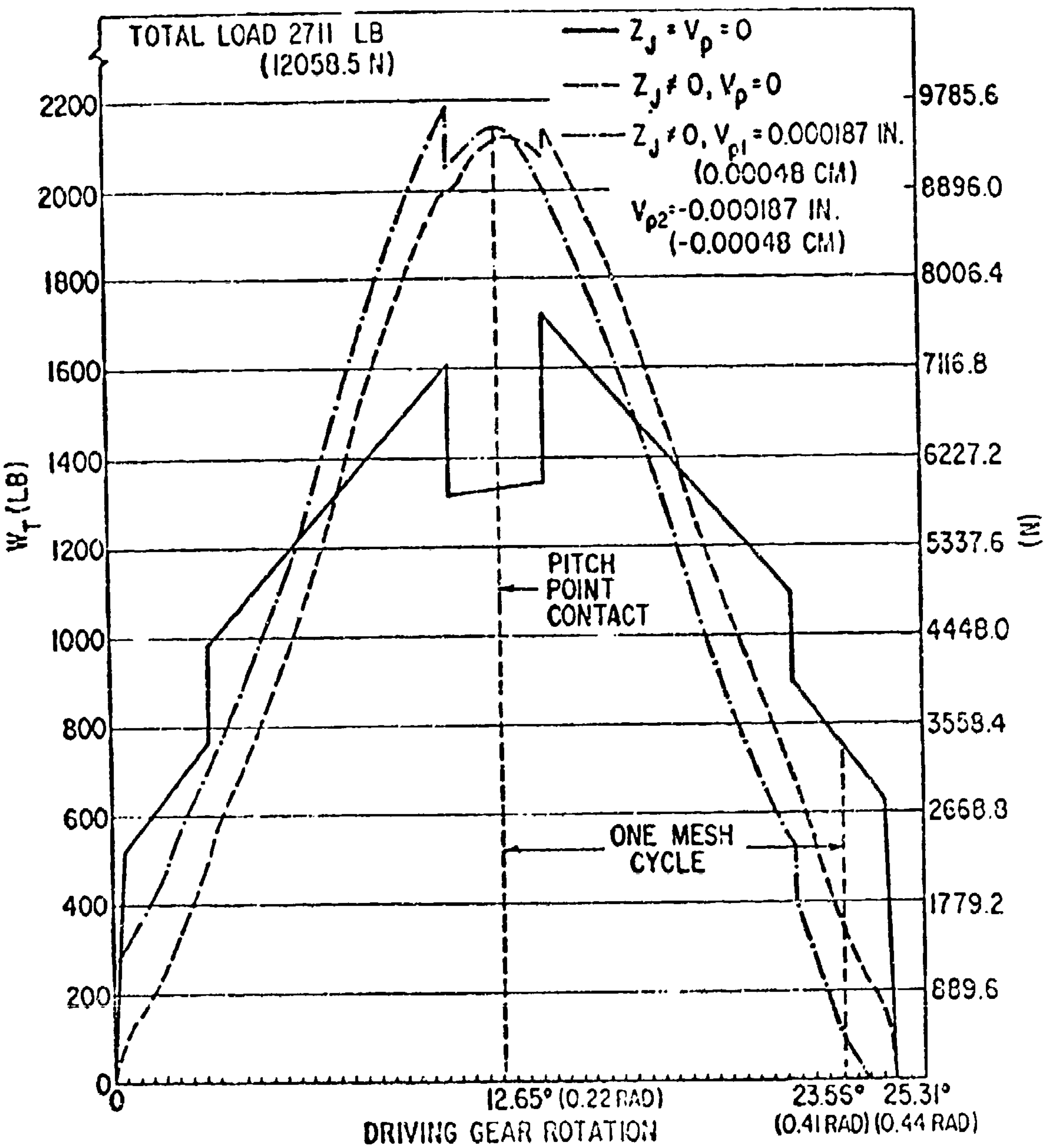


Figure 16. Tangential Tooth Load.

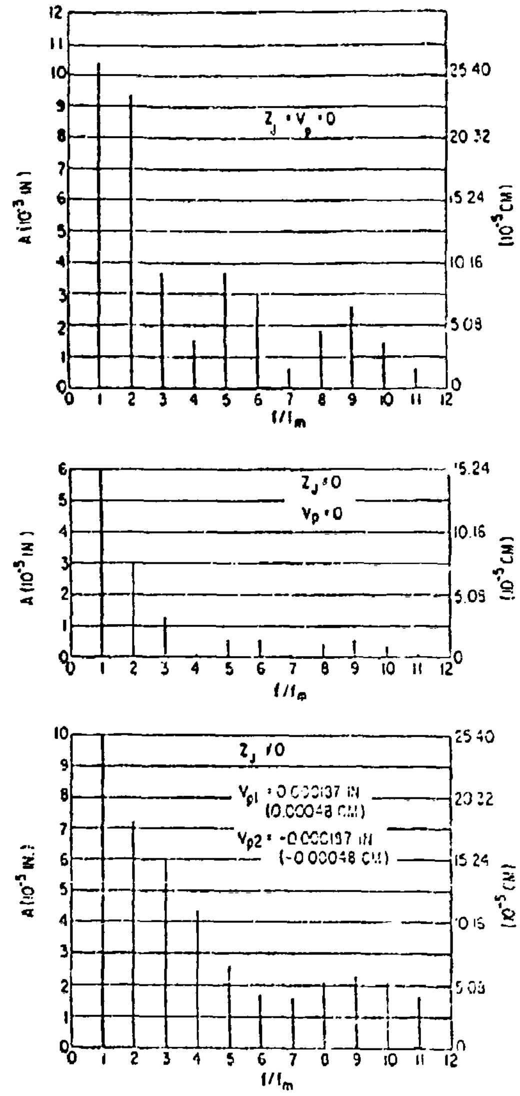


Figure 17. Gear Vibration Frequency Spectra.

The gears with the tooth profile modification still have three pairs of teeth in contact at the beginning and end of the mesh cycle and two pairs in the middle portion of the cycle as seen from Figure 15 . However, the transition when a tooth pair comes in and out of engagement is smoother with the profile modifications. The pitch-line tooth spacing errors decrease the total time that a tooth shares the tangential load relative to cases with perfect involute profiles. This is observed in Figure 16 . The effect of the spacing errors on peak tooth load is small, although the spacing errors modify the load sharing transition among engaging tooth pairs.

The frequency spectra of the vibration induced by gear mesh for all cases are discrete as shown in Figure 17. The amplitude is non-zero only at integer multiples of the mesh frequency. This is because all the teeth in each of the meshing gears are assumed identical, and all other gear parameters such as center distance and tangential tooth load are assumed constant. The only characteristic frequency is the mesh frequency. The spectrum of the vibration associated with the gear mesh thus contains signals at only the mesh frequency and its higher harmonics. If the gear parameters change or if the teeth in a gear are not identical the tooth meshing error pattern will, in general, vary between mesh cycles. The associated vibration will then contain signal components at the so-called sideband frequencies, not being the mesh frequency harmonics. This fact has been extensively discussed in Reference 14. Figure 17 shows that the vibration amplitude in general decreases with increasing frequency. The frequency spectra of the three cases shown in Figure 17 have the largest amplitudes at the first harmonic. The tooth profile modification appears to decrease the vibration signals at all harmonic frequencies, although the mean tooth deflection is calculated to be larger with the profile modification. The tooth spacing error is seen to have the effect of raising the signals at all frequencies. The amplitudes at some high harmonics are even larger than those of the case with perfect tooth profiles and no spacing error.

SUMMARY

A method has been developed for predicting the level and character of gear-mesh-induced vibrations in high-contact-ratio spur gears. In the computation of vibration amplitude, the load sharing among pairs of engaging gear teeth is calculated by considering all modes of tooth compliance, as well as any tooth profile modification and tooth spacing error.

Mesh dynamics calculations have been performed for a set of high-contact-ratio spur gears. It was found that the frequency spectrum of the gear-mesh-induced vibration depends strongly upon the tooth profile modification and the spacing error. The profile modification can effectively change the contact ratio of the gears by altering the maximum number of tooth pairs in contact simultaneously.

This analysis is useful for designing low-vibration gears by properly controlling relevant gear parameters such as tooth profile modifications. This approach may also be used to control peak tooth loads.

COMPUTER PROGRAM (R-75) INPUT FORMAT

The mesh dynamics analysis for high-contact-ratio spur gears has been programmed in computer program GGEAR/HCR (R-75), and a listing is provided in Appendix B. The limitations and assumptions involved in the analysis were stated previously. The user specifies a pair of geometrically compatible spur gears and their operating conditions; the program computes the load sharing among pairs of teeth, the stiffness of individual tooth pairs, the tooth displacement deviation, and the frequency spectrum of the associated vibration.

Figure 18 shows a sample input, and following is a brief discussion of the input variables, format, and important instructions for setting up the input data required for program R-75.

Input Variables, Format, and Instructions

Card 1 Title, columns 2 through 72.

Card 2 Control numbers. Format (16I5).

a. NMC Number of mesh cycles (last digit in column 5).

b. INT Specifies if this control card represents the last complete set of input data being submitted.

If more sets of input data follow use 0.

If this is the last set use 1 (column 10).

c. MN Specifies the types of spur gears considered.

If both the driving and driven gears are external gears use 1.

If the driving gear is an external gear and if the driven gear is an internal (ring) gear use 0 (The program will not run properly if the internal gear is submitted as the driving gear.) (column 15).

d. MMM Number of initial terms of the Fourier analysis for which coefficients will be printed, beyond the coefficient for the constant term. This number cannot exceed (FI x NMC + NMC/2) where FI is the input variable submitted on Card 4 (last digit in column 20).

```
1    BOEING VERTOL     H-C-R    SPUR GEAR             TEST CASE 2
     1     1     1    12     0     1     0     1     1
     0     0     0     0     0     1     0
33.            58.            4.55           21.            3.950          6.672
3.297          6.019          12.            .3685
.3140          0.5            0.5            0.087          0.083
10.            +.000187       -.000187
     1
9831.          9831.          9831.          9831.          9831.          9831.
9831.          9831.          9831.          9831.          9831.          9831.
9831.          9831.          9831.          9831.          9831.          9831.
9831.          9831.          9831.          9831.          9831.          9831.
9831.
-.00095        -.00008        -.00009        -.00075        27.            18.
-.00075        -.0061          -.0003         -.00055        24.56          16.65

0000000000000000000000
```

Figure 18. Sample R-75 Input Deck.

e. IPLT Specifies if the calculated tooth mesh error is to be plotted.

If tooth meshing error is to be plotted use 1.

If plotting is to be bypassed use 0 (column 25).

f. IFOUR Specifies if a Fourier analysis of the tooth mesh error is to be performed.

If it is to be performed use 1.

If it is to be bypassed use 0 (column 30).

g. ISPECT Specifies if a power spectral density function for the tooth mesh error is to be calculated.

If it is to be calculated use 1 and input NWS and WS on Card 9.

If it is to be bypassed use 0 (column 30).

h. IHICR Specifies if the pair of gears is high-contact-ratio.

If the contact ratio of the gears is less than 2 use 0.

If the contact ratio is greater than 2 use 1.

Note that the contact ratio has to be less than 4 (column 35).

Card 2a Option Card. Format (16I5)

a. IRM Specifies if tooth depth or root radius is input.

If IRM = 0 omit Card 4a and root radius is input.

If IRM = 1 submit Card 4a and tooth depth is input (column 5).

b. IPERC Specifies if circular tooth thickness or a percentage quantity which allocates to drive and driven gears the difference between the actual circular pitch and the working backlash is input.

If this is 0 omit Card 5a and input the circular tooth thickness.

If this is 1 submit Card 5a on which the backlash and percent split to driving member are input (column 10).

c. IRF Specifies if the fillet radius is input.

If it is 0 fillet radii RF1 and RF2 are input.

It it is 1 RF1 and RF2 are calculated based on a circular fillet (column 15).

d. IPROP Specifies if the gear material properties are input.

If it is 0 omit Card 6 and steel properties are provided internally.

If it is 1 submit Card 6 for material properties (column 20).

e. ICENT Specifies how center distance is input.

If it is 0 input center distance on Card 7 and omit variable DEV on Card 7.

If it is 1 omit variable CL on Card 7 and input DEV (deviation from standard center distance) on Card 7.

If it is 2, standard center distance is used and omit variables CL and DEV on Card 7 (column 25).

f. NZJ Specifies how the tooth profile deviation and tooth support compliance are to be input, according to the following table:

If	Submit					
NZJ	U pt-by-pt	On Card	Z pt-by-pt	On Card	4 Zvalues	On Card
0	Yes	8a	Yes	8b	no	
1	No		No		Yes	8d
2	No		No		No	
3	No		Yes	8b	No	
4	Yes	8a	No		Yes	8d
5	Yes	8a	No		No	

In the Table U is the tooth support compliance explained in the description of Card 8a. If U is not input (NZJ = 1, 2, or 3) a value of 0 is used internally. Z is the deviation of points on the actual tooth profile from a true involute. Z is explained in the description of Cards 8b and 8d (column 30).

g. IDIAG Specifies if diagnostic information is to be printed.

If it is 0 no diagnostics are printed.

If it is 1 diagnostics are printed out (column 35).

Card 3 Gear Design Data Format (6E 13.5).

a. FNI Number of teeth, driving gear. Use columns 1 through 13.

b. FN2 Number of teeth, driven gear. Use columns 14 through 26.

c. DPIT Diametral pitch, 1/in. (columns 27 through 39).

d. PANG Cutting pressure angle, deg (columns 40 through 52).

e. RØ1 Radius to the outside diameter of the driving gear, in. This should be reduced by any radial loss in working surface at the tip of the teeth from rounding or chamfering (columns 53 through 65)

f. RØ2 or RI2 Radius to the outside diameter of the driven gear, if external, and to the inside diameter, if internal, in. This should be corrected for any radial loss in working surface at the tip of the teeth from tip rounding or chamfering. In the case of an internal gear, this radius must be equal to or greater than the base circle radius. No check for this is provided (columns 66 through 78)

Card 4 Gear design data, continued Format (6E 13.5)

a. RM1 Radius to the tooth circle of the driving gear, in. If this radius is smaller that the computed base circle radius the input value of root radius is used at some points in the program and this is noted in the output. If the root radius is smaller than the base circle radius so that the root fillet center lies inside the base circle, the tooth outline between the base circle and the fillet is assumed to be a radial line by the program. Omit if IRM $\neq$ 0 (columns 1 through 13)

b. RM2 Radius to the root circle of the driven gear, in. For the case of an external gear, the same comments as above apply. Omit if IRM $\neq$ 0 (columns 14 through 26)

c. FI Number which establishes the number of calculation points (N_J) where $N_J = 1 + 2$ (FI). The calculation points may be viewed as selected contact points on the true involute profile, extended where necessary. These contact points with the mating involute area are associated with specific angular positions of the gear as it is rotated, where the angular positions correspond to uniform subdivisions of the tooth spacing angle. A greater number of these points will give more closely spaced point-by-point output data and more accurate calculations of tooth deflections and Fourier coefficients. A value of FI = 12, giving 25 calculation points, has been found to be convenient. (Maximum value of FI = 12.) (Columns 27 through 39)

d. T1 Circular tooth thickness at the pitch circle, driving gear, in. Omit if IPERC $\neq$ 0 (columns 40 through 52).

Card 4a Gear Design Data, Continued Format (6E 13.5)

This card is needed only if IRM $\neq$ 0.

a. HT1 Tooth depth, driving gear, in. (columns 1 through 13).

b. HT2 Tooth depth, driven gear, in. (columns 14 through 26).

Card 5 Gear Design Data, Continued Format (5E 13.5)

a. T2 Circular tooth thickness at the pitch circle, driven gear, in. Omit if IPERC $\neq$ 0 (columns 1 through 13).

b. F1 Effective tooth face width, driving gear, in. If the face widths of the mating gears are similar, use the actual face width. If one tooth is much wider than its mate use an effective face width which allows for the limited additional support that the greater width provides (columns 14 through 26).

c. F2 Effective tooth face width, driven gear, in. The comments for F1 also apply here (columns 27 through 39).

d. RF1 Fillet radius, driving gear, in. Omit if IRF $\neq$ 0 (columns 40 through 52).

e. RF2 Fillet radius, driven gear, in. Omit if IRF $\neq$ 0 (columns 53 through 65).

Card 5a Tooth Thickness Parameter Format (6E 13.5)

This card is needed only if IPERC = 1.

BL Backlash (in.) (columns 1 through 13).

PT1 Percent tooth thickness split to driving member (columns 14 through 26).

Card 6 Gear Design Data, Continued Format (6E 13.5)

This card is needed only if IPROP = 1.

If IPROP = 0, material is assumed to be steel.

a. YE1 Young's modulus (in bending) driving gear, lb/in^2 (columns 1 through 13).

b. YE2 Young's modulus (in bending) driven gear, lb/in^2 (columns 14 through 26).

c. GE1 Shear modulus, driving gear, lb/in^2 (columns 27 through 39).

d. GE2 Shear modulus, driven gear, lb/in^2 (columns 40 through 52).

e. PØS1 Poisson's ratio, driving gear (columns 53 through 65).

Since this ratio is used only in the allowance for the "wide beam effect", it should be reduced for cases where tooth face width is not much greater than tooth thickness with a limiting value of zero when the tooth width is smaller than the thickness (columns 53 through 65).

f. PØS2 Poisson's ratio, driven gear. Comments for PØS1 also apply here (columns 66 through 78).

There are NMC sets of the following Card 7 and Card 8. Each set supplies center distance, tooth spacing errors, tooth profile errors, tooth support compliances and transmitted torque for one mesh cycle.

Card 7 Center distance and tooth spacing error data. Format (3E 13.5)

a. CL Center distance, in.

This is the actual center distance including any spreading under load. Omit if ICENT $\neq$ 0 (columns 1 through 13).

b. VPT1 Tooth spacing error, driving gear, in. This error is based on the distance between the pitch points of successive teeth, adjusted to the direction of the line of action. This adjustment is accomplished by multiplying the pitch line error by the cosine of the pressure angle. The error is positive if the measured spacing is smaller than the desired spacing (columns 14 through 26).

c. VPT2 Tooth spacing error, driven gear, in. The comments under VPT1 also apply here (columns 27 through 39).

d. DEV Deviation from standard center distance. Omit if ICENT $\neq$ 1 (columns 40 through 52).

The total number of Cards 8a, 8b, 8c and 8d depends on the number of calculation points (N_J) (see Card 4). The total number of points along the tooth profile (M) at which the tooth profile deviation ZJ and the tooth support compliance UJ must be specified is $M = 2N_J$. If IHICR = 0 unless it is a high contact ratio mesh (IHICR = 1) in which case $M = 4N_J$. When the point-by-point option is used to input the values of ZJ (Cards 8b-1 and 8b-2) and UJ (Cards 8a-1 and 8a-2), the specification of these values is described below. A value of ZJ or UJ must be introduced even if there is no contact at the particular calculation point or if the tooth profile does not actually extend to the calculation point. A blank space may be used for these points.

For the driving gear, the first value is for the calculation point located (M/2-1) points preceding the pitch point (or inside the pitch circle); the (M/2)th value is for the pitch point; the last, or (M)th, value is for the calculation point located (M/2) points after the pitch point (or outside the pitch circle). This last point may also be described as the point of contact on one meshing tooth when the point of contact on the next meshing tooth is the pitch point.

For an external driven gear, the first value is for the calculation point located (M/2) points before the pitch point (or inside the pitch circle); the (M/2 + 1)th value is for the pitch point; the last, or (M)th, value is for the calculation point located (M/2 - 1) points after the pitch point (or outside the pitch circle). The first point may also be described as the point of contact on one meshing tooth when the point of contact on the previous meshing tooth is the pitch point.

For an internal driven gear, the first value is for the calculation point located (M/2) points following the pitch point (or outside the pitch circle); the (M/2 + 1)th value is for the pitch point; the last, or (M)th, value is for the calculation point located (M/2 - 1) points before the pitch point (or inside the pitch circle). The first point may also be described as the point of contact on the meshing tooth when the point of contact on the next meshing tooth is the pitch point.

Card 8a-1 Tooth support compliance, driving gear. Format (6E 13.5)

This group of cards is needed only if NZJ = 0, 4 or 5.

UJ1 Tooth support compliance or any compliance supplementary to the tooth compliance included in the analysis for the driving gear, in/lb. This compliance is the deflection normal to the profile under unit load at the calculation point on the profile. A uniform compliance for all calculation points, such as would result from a uniform gear shaft compliance, will not affect motion irregularities or load transfer. It will, however, increase the mean deviation in transmitted motion. There are M values required.

Card 8a-2 Tooth support compliance, driven gear. Format (6E 13.5)

This group of cards is needed only if NZJ = 0, 4 or 5.

UJ2 Tooth support compliance for the driven gear, in/lb. The comments under UJ1 also apply here.

Card 8b-1 Tooth profile deviation data, driving gear. Format (6E 13.5)

This group of cards is needed only if NZJ = 0 or 3.

ZJ1 Deviation from a true involute, measured normal to the profile, of the point on the actual tooth profile of the driving gear, in. This true involute is positioned relative to the actual profile so that its deviation at the pitch point is zero. Where the deviation represents material added to the true involute, it is positive; where it represents material subtracted, it is negative. If the profile does not extend to a particular calculation point or if the mating gear will not contact at this point, the deviation may be input as zero. There are M values required.

Card 8b-2 Tooth profile deviation data, driven gear. Format (6E 13.5)

This group of cards is needed only if NZJ = 0 or 3.

ZJ2 Deviation from a true involute of the point on the actual tooth profile of the driven gear, in. The comments under ZJ1 also apply here.

Card 8c-1 Torque. Format (16I5)

NTS Specifies if the torque input on the next group of cards applies to the driving or driven gears, 1 = driving gear, 2 = driven gear.

Card 8c-2 Transmitted torque. Format (6E13.5)

TØR Torque values of driving or driven gear, in/lb. A variable torque through one mesh cycle is allowed. There are N_J values required for the N_J calculation points. The first calculation point corresponds to the first data point outside the pitch circle of the driving gear. In the case of constant torque N_J equal values should be input.

In specifying the deviation data, the true involute is positioned relative to the actual profile so that its deviation at the pitch point is zero. Where the deviation represents material added to the true involute, it is positive; where it represents material subtracted, it is negative. The roll angles for the high and low points must be specified. The tooth profile deviations at all other calculation points are obtained internally by linear interpolation.

Card 8d-1 Tooth profile deviation, driving gear. Format (6E13.5).

This card is needed only if NZJ = 1 or 4.

a. Z(1,1) Deviation of the outer diameter point on the actual tooth profile from the true involute, in. (columns 1 through 13).

b. Z(5,1) Deviation of the high point (between outer diameter point and pitch point) on the actual tooth profile from the true involute, in. (columns 40 through 52).

c. Z(4,1) Deviation of the low point (between pitch point and the first contact point near the tooth root) on the actual tooth profile from the true involute, in. (columns 27 through 39).

d. Z(3,1) Deviation of the true involute point (the first contact point) on the actual tooth profile from the true involute, in. (columns 40 through 52).

e. RHIGH(1) Roll angle of the high point, deg. (columns 53 through 65).

f. RLOW(1) Roll angle of the low point, deg. (columns 66 through 78).

Card 8d-2 Tooth profile deviation, driven gear. Format (6E13.5).

This card is needed only if NZJ = 1 or 4.

a. Z(1,2) Deviation of the outer diameter point for an external gear. For an internal gear, it is the deviation of the inner diameter point, in. (columns 1 through 13).

b. Z(5,2) Deviation of the high point (between inner diameter and pitch point for internal gear), in. (columns 14 through 26).

c. Z(4,2) Deviation of the low point, in. (columns 27 through 39).

d. Z(3,2) Deviation of the true involute point, in. (columns 45 through 52).

e. RHIGH(2) Roll angle of the high point, deg. (columns 53 through 65).

f. RLOW(2) Roll angle of the low point, deg. (columns 66 through 78).

Card 9 Gear speed. Format (I5, E 13.5)

a. NWS Identifies if the input speed is the speed of driving or driven gear. 1 = driving gear, 2 = driven gear (column 5).

b. WS Driving or driven gear speed, rpm (columns 6 through 18).

PROGRAM OUTPUT AND DESCRIPTION

Figure 19 shows a sample output for the case defined in Figure 18. Following is a brief discussion on the output data.

Output Variables and Explanations

1. Title

2. Control Number - NMC, INT, MN, MMM, IPLT, IFOUR, ISPECT, IHICR, IRM, IPERC, IRF, IPROP, ICENT, NZJ, as in input Cards 2 and 2a.

3. Design Data - FI(=I), FN1(=N1), DPIT(DIA. PITCH), RO1, RT1, RM1, BLANK, FN2(=N2), PANG(CUT. PRESS ANG), RO2 or RI2, RT2, RM2, T1, F1, RF1, YE1, GE1, PØS1, T2, F2, RF2, YE2, GE2, PØS2, as in input Cards 3 through 6.

Items 4 through 11 are printed for each of the NMC mesh cycles specified.

4. Mesh Cycle Identification, Center Distance and Tooth Spacing Errors.

5. Input Listing of Transmitted Torques at N_J Calculation Points and Listing of Calculated Tangential Tooth Loads at N_J Calculation Points.

6. Input Listing of Profile Error and Supplementary Compliance - ZJ1, ZJ2, UJ1, UJ2, as in input Cards 8a, 8b and 8d.

7. Operating Pressure Angle, deg.

8. Miscellaneous Data:

   RP1 pitch circle radius, driving gear, in.

   RP2 pitch circle radius, driven gear, in.

   RB1 base circle radius, driving gear, in.

   RB2 base circle radius, driven gear, in.

   BA1 arc of approach, driving gear, rad.

   BA2 arc of approach, driven gear, rad. (negative on internal gear)

   BR1 arc of recess, driving gear, rad.

   BR2 arc of recess, driven gear, rad. (negative on internal gear)

```
BOEING VERTOL   M-C-R   SPUR GEAR          TEST CASE 2

PN0335=COMPUTATIONS OF GEAR TOOTH MESHING ERRORS       3-22-1966

NMC  INT  MN   MMM  IPLT IFOUR ISPECT IWICH  IRM IFENC  IRF IPROP ICENT  NZJ
  1    1   1    12    0     1     0      1    0     0    0     0     0    1

       I            N1        DIA. PITCH       RO1          RT1          RM1
1.20000E+01  3.30000E+01  4.55000E+00  3.95000E+00  3.38551E+00  3.29700E+00

                    N2      CUT. PRESS ANG     RO2          RT2          RM2
             5.80000E+01  2.10000E+01  6.67200E+00  6.08415E+00  6.01900E+00

      T1           F1           RF1      YOUNGS MOD-1  SHEAR MOD-1  POS.RATIO-1
3.68500E-01  5.00000E-01  8.70000E-02  3.00000E+07  1.15000E+07  3.00000E-01

      T2           F2           RF2      YOUNGS MOD-2  SHEAR MOD-2  POS.RATIO-2
3.14000E-01  5.00000E-01  8.30000E-02  3.00000E+07  1.15000E+07  3.00000E-01

CALCULATED DATA

    *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *

MESH CIRCLE NO.=  1.   CENTER DISTANCE= 1.000000E+01 IN.
VPT1. TOOTH SPACING ERROR (DRIVING GEAR) = 1.870000E-04 IN.
VPT2. TOOTH SPACING ERROR (DRIVEN  GEAR) =-1.870000E-04 IN.

   DRIVING GEAR  TRANSMITTED TORQUE (IN-LB)
  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03
  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03
  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03  9.8310E+03

             TOTAL LOAD TANGENT TO PITCH CIRCLE, TRANSMITTED BY GEAR TEETH (LB)
  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03
  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03
  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03  2.7110E+03

             DEVIATION OF POINT ON ACTUAL TOOTH PROFILE FROM TRUE INVOLUTE
                                   DRIVING GEAR
               AT OUTER DIAM.   HIGH POINT     PITCH POINT    LOW  POINT - TRUE INVOLUTE

   Z              -9.50000E-04  -8.00000E-05  0.            -2.00000E-05  -7.50000E-04
   ROLL ANGLE      3.44183E+01   2.70000E+01  2.19938E+01   1.80000E+01   9.56988E+00

             DEVIATION OF POINT ON ACTUAL TOOTH PROFILE FROM TRUE INVOLUTE
                                   DRIVEN GEAR
               AT OUTER DIAM.   HIGH POINT     PITCH-POINT    LOW  POINT - TRUE INVOLUTE

   Z              -7.50000E-04  -1.00000E-04  0.            -3.00000E-04  -5.50000E-04
   ROLL-ANGLE     -2.50626E+01   2.45660E+01  2.19938E+01   1.66500E+01   1.49133E+01
 ANGLE O   PROACH ON DRIVING GEAR IS GREATER THAN TOOTH SPACING  GLE.
 PROGRAM CONTINUED WITHOUT OVERLAP

 ANGLE OF RECESS ON DRIVING GEAR IS NOT SMALLER THAN TOOTH SPACING ANGLE.
 PROGRAM CONTINUED WITHOUT OVERLAP

    RP1          RR1          RA1          RR1          AT1
 3.62637E+00  3.38551E+00  2.16838E-01  2.17197E-01  3.83864E-01

    RP2          RR2          RA2          RR2          AT2
 6.37363E+00  5.95029E+00  1.23578E-01  1.23373E-01  1.70565E-01

 DRIVING GEAR INPUT ROOT RADIUS SMALLER THAN BASE CIRCLE RADIUS

 DRIVING GEAR INPUT RADIUS TO FILLET CENTER INSIDE BASE CIRCLE.

 PROGRAM CONTINUES WITH CORRECT TREATMENT.

     J1           CJ1          AJ1         QJIABC        XJ1          YJ1          ZJ1
 -4.90000E+01  0.           -3.73183E-01
 -4.80000E+01  0.           -3.65567E-01
 -4.70000E+01  0.           -3.57951E-01
 -4.60000E+01  0.           -3.50335E-01
 -4.50000E+01  0.           -3.42719E-01
 -4.40000E+01  0.           -3.35103E-01
 -4.30000E+01  0.           -3.27487E-01
 -4.20000E+01  0.           -3.19871E-01
 -4.10000E+01  0.           -3.12255E-01
 -4.00000E+01  0.           -3.04639E-01
 -3.90000E+01  0.           -2.97023E-01
 -3.80000E+01  0.           -2.89407E-01
 -3.70000E+01  0.           -2.81791E-01
 -3.60000E+01  0.           -2.74175E-01
 -3.50000E+01  0.           -2.66559E-01
 -3.40000E+01  0.           -2.58943E-01
 -3.30000E+01  0.           -2.51327E-01
 -3.20000E+01  0.           -2.43711E-01
 -3.10000E+01  0.           -2.36095E-01
 -3.00000E+01  0.           -2.28479E-01
 -2.90000E+01  0.           -2.20863E-01
 -2.80000E+01  1.00000E+00  -2.13248E-01  7.71877E-08  3.42684E+00  2.28311E-01  -7.33894E-04
 -2.70000E+01  1.00000E+00  -2.05632E-01  8.00531E-08  3.43131E+00  2.27834E-01  -5.99730E-04
 -2.60000E+01  1.00000E+00  -1.98016E-01  8.30919E-08  3.43597E+00  2.27300E-01  -6.65567E-04
 -2.50000E+01  1.00000E+00  -1.90400E-01  8.63099E-08  3.44083E+00  2.26707E-01  -6.31404E-04
```

Figure 19. R-75 Computer Output.

-2.40000E+01	1.00000E+00	-1.92764E-01	8.97138E-08	3.44587E+00	2.26057E-01	-5.97241E-04
-2.30000E+01	1.00000E+00	-1.75168E-01	9.33104E-08	3.45110E+00	2.25332E-01	-5.63077E-04
-2.20000E+01	1.00000E+00	-1.67552E-01	9.71075E-08	3.45653E+00	2.24544E-01	-5.28914E-04
-2.10000E+01	1.00000E+00	-1.59936E-01	1.01113E-07	3.46214E+00	2.23684E-01	-4.94751E-04
-2.00000E+01	1.00000E+00	-1.52320E-01	1.05337E-07	3.46793E+00	2.22751E-01	-4.60588E-04
-1.90000E+01	1.00000E+00	-1.44704E-01	1.09789E-07	3.47392E+00	2.21741E-01	-4.26425E-04
-1.80000E+01	1.00000E+00	-1.37088E-01	1.14480E-07	3.48009E+00	2.20651E-01	-3.92261E-04
-1.70000E+01	1.00000E+00	-1.29472E-01	1.19420E-07	3.48644E+00	2.19479E-01	-3.58098E-04
-1.60000E+01	1.00000E+00	-1.21856E-01	1.24624E-07	3.49298E+00	2.18221E-01	-3.23935E-04
-1.50000E+01	1.00000E+00	-1.14240E-01	1.30104E-07	3.49970E+00	2.16875E-01	-2.89772E-04
-1.40000E+01	1.00000E+00	-1.06624E-01	1.35874E-07	3.50660E+00	2.15438E-01	-2.55608E-04
-1.30000E+01	1.00000E+00	-9.90078E-02	1.41943E-07	3.51368E+00	2.13987E-01	-2.21445E-04
-1.20000E+01	1.00000E+00	-9.13918E-02	1.48351E-07	3.52095E+00	2.12279E-01	-1.87282E-04
-1.10000E+01	1.00000E+00	-8.37758E-02	1.55043E-07	3.52839E+00	2.10551E-01	-1.53119E-04
-1.00000E+01	1.00000E+00	-7.61598E-02	1.62196E-07	3.53601E+00	2.08721E-01	-1.18956E-04
-9.00000E+00	1.00000E+00	-6.85438E-02	1.69681E-07	3.54380E+00	2.06785E-01	-8.85011E-05
-8.00000E+00	1.00000E+00	-6.09279E-02	1.77571E-07	3.55177E+00	2.04741E-01	-7.86676E-05
-7.00000E+00	1.00000E+00	-5.33119E-02	1.85890E-07	3.55992E+00	2.02587E-01	-6.88342E-05
-6.00000E+00	1.00000E+00	-4.56959E-02	1.94684E-07	3.56823E+00	2.00319E-01	-5.98007E-05
-5.00000E+00	1.00000E+00	-3.80799E-02	2.03922E-07	3.57672E+00	1.97934E-01	-4.91673E-05
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
-.00000E	1.00000E+00	-2.28479E-02	2.24011E-07	3.59421E+00	.92805E-01	-2.95004E-05
-2.00000E	1.00000E+00	-1.52320E-02	2.34911E-07	3.60321E+00	.90055E-01	-1.96669E-05
-1.00000E+00	1.00000E+00	-7.61598E-03	2.44430E-07	3.61237E+00	1.87178E-01	-9.83345E-06
0.	1.00000E+00	0.	2.58611E-07	3.62169E+00	1.84171E-01	0.
1.00000E+00	1.00000E+00	7.61598E-03	2.71496E-07	3.63118E+00	1.81031E-01	-6.97316E-06
2.00000E+00	1.00000E+00	1.52320E-02	2.85135E-07	3.64084E+00	1.77756E-01	-1.39463E-05
3.00000E+00	1.00000E+00	2.28479E-02	2.99540E-07	3.65065E+00	1.74343E-01	-2.09195E-05
4.00000E+00	1.00000E+00	3.04639E-02	3.14887E-07	3.66062E+00	1.70789E-01	-2.78926E-05
5.00000E+00	1.00000E+00	3.80799E-02	3.31119E-07	3.67075E+00	1.67092E-01	-3.48658E-05
6.00000E+00	1.00000E+00	4.56959E-02	3.48343E-07	3.68103E+00	1.63250E-01	-4.18389E-05
7.00000E+00	1.00000E+00	5.33119E-02	3.66633E-07	3.69147E+00	1.59259E-01	-4.88121E-05
8.00000E+00	1.00000E+00	6.09279E-02	3.86071E-07	3.70205E+00	1.55117E-01	-5.57853E-05
9.00000E+00	1.00000E+00	6.85438E-02	4.06746E-07	3.71279E+00	1.50822E-01	-6.27584E-05
1.00000E+01	1.00000E+00	7.61598E-02	4.28753E-07	3.72368E+00	1.46371E-01	-6.97316E-05
1.10000E+01	1.00000E+00	8.37758E-02	4.52218E-07	3.73472E+00	1.41761E-01	-7.67047E-05
1.20000E+01	1.00000E+00	9.13918E-02	4.77248E-07	3.74590E+00	1.36990E-01	-1.06919E-04
1.30000E+01	1.00000E+00	9.90078E-02	5.03988E-07	3.75722E+00	1.32056E-01	-1.57958E-04
1.40000E+01	1.00000E+00	1.06624E-01	5.32593E-07	3.76869E+00	1.26956E-01	-2.08996E-04
1.50000E+01	1.00000E+00	1.14240E-01	5.63243E-07	3.78029E+00	1.21688E-01	-2.60034E-04
1.60000E+01	1.00000E+00	1.21856E-01	5.96142E-07	3.79203E+00	1.16249E-01	-3.11073E-04
1.70000E+01	1.00000E+00	1.29472E-01	6.31526E-07	3.80391E+00	1.10636E-01	-3.62111E-04
1.80000E+01	1.00000E+00	1.37088E-01	6.69673E-07	3.81593E+00	1.04849E-01	-4.13150E-04
1.90000E+01	1.00000E+00	1.44704E-01	7.10908E-07	3.82807E+00	9.88828E-02	-4.64188E-04
2.00000E+01	1.00000E+00	1.52320E-01	7.55625E-07	3.84034E+00	9.27369E-02	-5.15226E-04
2.10000E+01	1.00000E+00	1.59936E-01	8.04279E-07	3.85275E+00	8.64004E-02	-5.66265E-04
2.20000E+01	1.00000E+00	1.67552E-01	8.57523E-07	3.86527E+00	7.98950E-02	-6.17303E-04
2.30000E+01	1.00000E+00	1.75168E-01	9.16043E-07	3.87792E+00	7.31945E-02	-6.68341E-04
2.40000E+01	1.00000E+00	1.82784E-01	9.80832E-07	3.89070E+00	6.63047E-02	-7.19380E-04
2.50000E+01	1.00000E+00	1.90400E-01	1.05319E-06	3.90359E+00	5.92232E-02	-7.70418E-04
2.60000E+01	1.00000E+00	1.98016E-01	1.13493E-06	3.91660E+00	5.19479E-02	-8.21457E-04
2.70000E+01	1.00000E+00	2.05632E-01	1.22872E-06	3.92972E+00	4.44765E-02	-8.72495E-04
2.80000E+01	1.00000E+00	2.13248E-01	1.33869E-06	3.94296E+00	3.68070E-02	-9.23533E-04
2.90000E+01	0.	2.20863E-01				
3.00000E+01	0.	2.28479E-01				
3.10000E+01	0.	2.36095E-01				
3.20000E+01	0.	2.43711E-01				
3.30000E+01	0.	2.51327E-01				
3.40000E+01	0.	2.58943E-01				
3.50000E+01	0.	2.66559E-01				
3.60000E+01	0.	2.74175E-01				
3.70000E+01	0.	2.81791E-01				
3.80000E+01	0.	2.89407E-01				
3.90000E+01	0.	2.97023E-01				
4.00000E+01	0.	3.04639E-01				
4.10000E+01	0.	3.12255E-01				
4.20000E+01	0.	3.19871E-01				
4.30000E+01	0.	3.27487E-01				
.00000E+01	0.	3.35103E-01				
4.50000E+01	0.	3.42719E-01				
4.60000E+01	0.	3.50335E-01				
4.70000E+01	0.	3.57951E-01				
4.80000E+01	0.	3.65567E-01				
4.90000E+01	0.	3.73183E-01				
5.00000E+01	0.	3.80799E-01				

COOD. OF EFFECTIVE TOOTH PROFILE AT ROOT CIRCLE X Y

-3.28619E+00-2.66772E-01

J1	QJIA	QJIB	QJIC
-2.80000E+01	-3.15186E-09	5.86678E-08	1.53681E-08
-2.70000E+01	3.38309E-09	6.05939E-08	1.60761E-08
-2.60000E+01	3.64231E-09	6.25972E-08	1.68523E-08
-2.50000E+01	3.93223E-09	6.46773E-08	1.77004E-08
-2.40000E+01	4.25591E-09	6.68337E-08	1.86241E-08
-2.10000E+01	[illegible]	[illegible]	[illegible]
-2.00000E+01	4.96207E-09	7.62145E-08	2.11610E-08
-1.90000E	6.51353E-09	7.87464E-08	2.45289E-08
-1.80000E	7.12555E-09	8.13517E-08	2.60006E-08
-1.70000E+01	7.89474E-09	8.40349E-08	2.75813E-08
-1.60000E+01	8.55671E-09	8.67904E-08	2.92767E-08
-1.50000E+01	9.39613E-09	8.96294E-08	3.10929E-08
-1.40000E+01	1.03130E-08	9.25250E-08	3.30359E-08
-1.30000E+01	1.13342E-08	9.55045E-08	3.51121E-08
-1.20000E+01	1.24637E-08	9.85594E-08	3.73279E-08
-1.10000E+01	1.37123E-08	1.01690E-07	3.96903E-08
-1.00000E+01	1.50919E-08	1.04898E-07	4.22062E-08
-9.00000E+00	1.66153E-08	1.08183E-07	4.48928E-08
-8.00000E+00	1.82967E-08	1.11546E-07	4.77276E-08
-7.00000E+00	2.01516E-08	1.14985E-07	5.07481E-08
-6.00000E+00	2.21967E-08	1.18515E-07	5.39522E-08
-5.00000E+00	2.44504E-08	1.22123E-07	5.73479E-08
-4.00000E+00	2.69327E-08	1.25817E-07	6.09434E-08
-3.00000E+00	2.96655E-08	1.29598E-07	6.47472E-08
-2.00000E+00	3.26726E-08	1.33470E-07	6.87680E-08
-1.00000E+00	3.59809E-08	1.37436E-07	7.30144E-08
0.	3.95144E-08	1.41499E-07	7.74955E-08
1.00000E+00	4.36127E-08	1.45663E-07	8.22207E-08

2.00000E+00	4.80030E-08	1.49933E-07	8.71991E-08
3.00000E+00	5.23246E-08	1.54315E-07	9.24404E-08
4.00000E+00	5.[illegible]1197E-08	1.58814E-07	9.79543E-08
5.00000E+00	6.39790E-08	1.63438E-07	1.03751E-07
6.00000E+00	7.03083E-08	1.68195E-07	1.09840E-07
7.00000E+00	7.73082E-08	1.73093E-07	1.16232E-07
8.00000E+00	8.49964E-08	1.78144E-07	1.22937E-07
9.00000E+00	9.34215E-08	1.83356E-07	1.29966E-07
1.00000E+01	1.02478E-07	1.88751E-07	1.37340E-07
1.10000E+01	1.12241E-07	1.94338E-07	1.45039E-07
1.20000E+01	1.24096E-07	2.00137E-07	1.53105E-07
1.30000E+01	1.36277E-07	2.06171E-07	1.61539E-07
1.40000E+01	1.49776E-07	2.12455E-07	1.70353E-07
1.50000E+01	1.64634E-07	2.19049E-07	1.79554E-07
1.60000E+01	1.81070E-07	2.25959E-07	1.89163E-07
1.70000E+01	1.99104E-07	2.33235E-07	1.99183E-07
1.80000E+01	2.19102E-07	2.40941E-07	2.09630E-07
1.90000E+01	2.41265E-07	2.49124E-07	2.20515E-07
2.00000E+01	2.65992E-07	2.57843E-07	2.31850E-07
2.10000E+01	2.93347E-07	2.67204E-07	2.43645E-07
2.20000E+01	3.24087E-07	2.77522E-07	2.55921E-07
2.30000E+01	3.58656E-07	2.88706E-07	2.68691E-07
2.40000E+01	3.97867E-07	3.01044E-07	2.81942E-07
2.50000E+01	4.42594E-07	3.14911E-07	2.95716E-07
2.60000E+01	4.94096E-07	3.30523E-07	3.10015E-07
2.70000E+01	5.54537E-07	3.49325E-07	3.24854E-07
2.80000E+01	6.26859E-07	3.71585E-07	3.40244E-07

Z	AJ2	QJ2ABC	XJ2	YJ2	ZJ2
-5.00000E+01	-2.16662E-01				
-4.90000E+01	-2.12328E-01				
-4.80000E+01	-2.07995E-01				
-4.70000E+01	-2.03662E-01				
-4.60000E+01	-1.99324E-01				
-4.50000E+01	-1.94995E-01				
-4.40000E+01	-1.90662E-01				
-4.30000E+01	-1.86329E-01				
-4.20000E+01	-1.81996E-01				
-4.10000E+01	-1.77662E-01				
-4.00000E+01	-1.73329E-01				
-3.90000E+01	-1.68996E-01				
-3.80000E+01	-1.64663E-01				
-3.70000E+01	-1.60329E-01				
-3.60000E+01	-1.55996E-01				
-3.50000E+01	-1.51663E-01				
-3.40000E+01	-1.47330E-01				
-3.30000E+01	-1.42997E-01				
-3.20000E+01	-1.38663E-01				
-3.10000E+01	-1.34330E-01				
-3.00000E+01	-1.29997E-01				
-2.90000E+01	-1.25664E-01				
-2.80000E+01	-1.21330E-01	6.20647E-08	6.14791E+00	2.22552E-01	-5.31467E-04
-2.70000E+01	-1.16997E-01	6.62511E-08	6.15456E+00	2.21044E-01	-4.95727E-04
-2.60000E+01	-1.12664E-01	7.06142E-08	6.16132E+00	2.19483E-01	-4.59987E-04
-2.50000E+01	-1.08331E-01	7.51635E-08	6.16818E+00	2.17866E-01	-4.24247E-04
-2.40000E+01	-1.03994E-01	7.99095E-08	6.17515E+00	2.16194E-01	-3.88507E-04
-2.30000E+01	-9.96643E-02	8.48631E-08	6.18221E+00	2.14465E-01	-3.52767E-04
-2.20000E+01	-9.53311E-02	9.00358E-08	6.18937E+00	2.12678E-01	-3.17027E-04
-2.10000E+01	-9.09979E-02	9.54406E-08	6.19664E+00	2.10833E-01	-2.92702E-04
-2.00000E+01	-8.66646E-02	1.01089E-07	6.20400E+00	2.08929E-01	-2.78764E-04
-1.90000E+01	-8.23314E-02	1.06996E-07	6.21147E+00	2.06964E-01	-2.64826E-04
-1.80000E+01	-7.79982E-02	1.13177E-07	6.21903E+00	2.04939E-01	-2.50887E-04
-1.70000E+01	-7.36649E-02	1.19648E-07	6.22669E+00	2.02849E-01	-2.36949E-04
-1.60000E+01	-6.93317E-02	1.26424E-07	6.23446E+00	2.00698E-01	-2.23011E-04
-1.50000E+01	-6.49985E-02	1.33524E-07	6.24232E+00	1.98483E-01	-2.09073E-04
-1.40000E+01	-6.06652E-02	1.40967E-07	6.25027E+00	1.96203E-01	-1.95135E-04
-1.30000E+01	-5.63320E-02	1.48773E-07	6.25833E+00	1.93857E-01	-1.81196E-04
-1.20000E+01	-5.19988E-02	1.56963E-07	6.26648E+00	1.91445E-01	-1.67258E-04
-1.10000E+01	-4.76655E-02	1.65562E-07	6.27473E+00	1.88965E-01	-1.53320E-04
-1.00000E+01	-4.33323E-02	1.74593E-07	6.28307E+00	1.86417E-01	-1.39382E-04
-9.00000E+00	-3.89991E-02	1.84082E-07	6.29151E+00	1.83799E-01	-1.25444E-04
-8.00000E+00	-3.46659E-02	1.94058E-07	6.30004E+00	1.81112E-01	-1.11506E-04
-7.00000E+00	-3.03326E-02	2.04551E-07	6.30867E+00	1.78353E-01	-9.75673E-05
-6.00000E+00	-2.59994E-02	2.15592E-07	6.31740E+00	1.75523E-01	-8.36292E-05
-5.00000E+00	-2.16662E-02	2.27217E-07	6.32622E+00	1.72620E-01	-6.96910E-05
-4.00000E+00	-1.73324E-02	2.39462E-07	6.33513E+00	1.69644E-01	-5.57526E-05
-3.00000E+00	-1.29997E-02	2.52367E-07	6.34413E+00	1.66592E-01	-4.18146E-05
-2.00000E+00	-8.66646E-03	2.65975E-07	6.35323E+00	1.63466E-01	-2.78764E-05
-1.00000E+00	-4.33323E-03	2.80333E-07	6.36241E+00	1.60264E-01	-1.39382E-05
0.	0.	2.95491E-07	6.37169E+00	1.56984E-01	0.
1.00000E+00	4.33323E-03	3.11502E-07	6.38106E+00	1.53627E-01	-9.67480E-06
2.00000E+00	8.66646E-03	3.28426E-07	6.39052E+00	1.50191E-01	-1.93496E-05
3.00000E+00	1.29997E-02	3.46326E-07	6.40007E+00	1.46675E-01	-2.90244E-05
4.00000E+00	1.73324E-02	3.65272E-07	6.40971E+00	1.43079E-01	-3.86992E-05
5.00000E+00	2.16662E-02	3.85341E-07	6.41944E+00	1.39402E-01	-4.83740E-05
6.00000E+00	2.59994E-02	4.06616E-07	6.42926E+00	1.35642E-01	-5.80488E-05
7.00000E+00	3.03326E-02	4.29189E-07	6.43916E+00	1.31850E-01	-6.77236E-05
8.00000E+00	3.46659E-02	4.53163E-07	6.44916E+00	1.27874E-01	-7.73984E-05
9.00000E+00	3.89991E-02	4.78649E-07	6.45923E+00	1.23863E-01	-8.70732E-05
1.00000E+01	4.33323E-02	5.05772E-07	6.46940E+00	1.19767E-01	-9.67480E-05
1.10000E+01	4.76655E-02	5.34673E-07	6.47965E+00	1.15585E-01	-1.23794E-04
1.20000E+01	5.19988E-02	5.65510E-07	6.48998E+00	1.11316E-01	-1.59636E-04
1.30000E+01	5.63320E-02	5.98470E-07	6.50040E+00	1.06959E-01	-1.95478E-04
1.40000E+01	6.06652E-02	6.33726E-07	6.51091E+00	1.02513E-01	-2.31319E-04
1.50000E+01	6.49985E-02	6.71541E-07	6.52149E+00	9.79780E-02	-2.67161E-04
1.60000E+01	6.93317E-02	7.12170E-07	6.53216E+00	9.33525E-02	-3.03003E-04
1.70000E+01	7.36649E-02	7.55926E-07	6.54292E+00	8.86360E-02	-3.38844E-04
1.80000E+01	7.79982E-02	8.03175E-07	6.55375E+00	8.38276E-02	-3.74686E-04
1.90000E+01	8.23314E-02	8.54332E-07	6.56467E+00	7.89265E-02	-4.10527E-04
2.00000E+01	8.66646E-02	9.09979E-07	6.57565E+00	7.39313E-02	-4.46369E-04
2.10000E+01	9.09979E-02	9.70695E-07	6.58673E+00	6.88431E-02	-4.82211E-04
2.20000E+01	9.53311E-02	1.03729E-06	6.59789E+00	6.36593E-02	-5.18052E-04
2.30000E+01	9.96643E-02	1.11077E-06	6.60912E+00	5.83797E-02	-5.53894E-04
2.40000E+01	1.03098E-01	1.19243E-06	6.62043E+00	5.30035E-02	-5.89736E-04
2.50000E+01	1.08331E-01	1.28400E-06	6.63182E+00	4.75294E-02	-6.25577E-04
2.60000E+01	1.12664E-01	1.38766E-06	6.64328E+00	4.19582E-02	-6.61419E-04
2.70000E+01	1.16997E-01	1.50743E-06	6.65482E+00	3.62876E-02	-6.97261E-04
2.80000E+01	1.21330E-01	1.64792E-06	6.66644E+00	3.05173E-02	-7.33104E-04

-3.00000E+01	1.29997E-01
3.10000E+01	1.34330E-01
3.20000E+01	1.38663E-01
3.30000E+01	1.42997E-01
3.40000E+01	1.47330E-01
3.50000E+01	1.51663E-01
3.60000E+01	1.55996E-01
3.70000E+01	1.60330E-01
3.80000E+01	1.64663E-01
3.90000E+01	1.68996E-01
4.00000E+01	1.73329E-01
-4.10000E+01	1.77662E-01
4.20000E+01	1.81996E-01
4.30000E+01	1.86329E-01
-4.40000E+01	1.90662E-01
4.50000E+01	1.94995E-01
-4.60000E+01	1.99329E-01
-4.70000E+01	2.03662E-01
4.80000E+01	2.07995E-01
4.90000E+01	2.12328E-01

COOD. OF EFFECTIVE TOOTH PROFILE AT ROOT CIRCLE

X	Y
6.012502E+00	2.798188E-01

J2	QJ2A	QJ2B	QJ2C
-2.80000E+01	1.31644E-09	5.35799E-08	7.16834E-09
-2.70000E+01	1.60914E-09	5.64507E-08	8.19126E-09
-2.60000E+01	1.93591E-09	5.93743E-08	9.30392E-09
-2.50000E+01	2.30163E-09	6.23516E-08	1.05103E-08
-2.40000E+01	2.71167E-09	6.53835E-08	1.18143E-08
-2.30000E+01	3.17189E-09	6.84709E-08	1.32203E-08
-2.20000E+01	3.68864E-09	7.16149E-08	1.47322E-08
-2.10000E+01	4.26901E-09	7.48167E-08	1.63543E-08
-2.00000E+01	4.92047E-09	7.80775E-08	1.80910E-08
-1.90000E+01	5.65128E-09	8.13986E-08	1.99466E-08
-1.80000E+01	6.47539E-09	8.47814E-08	2.19256E-08
-1.70000E+01	7.38751E-09	8.82277E-08	2.40324E-08
-1.60000E+01	8.41316E-09	9.17390E-08	2.62716E-08
-1.50000E+01	9.55870E-09	9.53171E-08	2.86480E-08
-1.40000E+01	1.08365E-08	9.89642E-08	3.11660E-08
-1.30000E+01	1.22598E-08	1.02682E-07	3.38306E-08
-1.20000E+01	1.38432E-08	1.06474E-07	3.66466E-08
-1.10000E+01	1.56022E-08	1.10341E-07	3.96188E-08
-1.00000E+01	1.75536E-08	1.14287E-07	4.27522E-08
-9.00000E+00	1.97154E-08	1.18314E-07	4.60518E-08
-8.00000E+00	2.21086E-08	1.22427E-07	4.95228E-08
-7.00000E+00	2.47533E-08	1.26627E-07	5.31701E-08
-6.00000E+00	2.76730E-08	1.30920E-07	5.69991E-08
-5.00000E+00	3.08924E-08	1.35309E-07	6.10150E-08
-4.00000E+00	3.44392E-08	1.39798E-07	6.52232E-08
-3.00000E+00	3.83443E-08	1.44394E-07	6.96283E-08
-2.00000E+00	4.26374E-08	1.49100E-07	7.42377E-08
-1.00000E+00	4.73543E-08	1.53924E-07	7.90550E-08
0.	5.25330E-08	1.58871E-07	8.40864E-08
1.00000E+00	5.82149E-08	1.63949E-07	8.93376E-08
2.00000E+00	6.44451E-08	1.69166E-07	9.48141E-08
-3.00000E+00	7.12772E-08	1.74531E-07	1.00522E-07
4.00000E+00	7.87535E-08	1.80052E-07	1.06466E-07
5.00000E+00	8.69454E-08	1.85742E-07	1.12654E-07
6.00000E+00	9.5[illegible]47E-08	1.91412E-07	1.19089E-07
7.00000E+00	1.05734E-07	1.97675E-07	1.25780E-07
8.00000E+00	1.16494E-07	2.03948E-07	1.32731E-07
-9.00000E+00	1.28253E-07	2.10447E-07	1.39949E-07
1.00000E+01	1.41142E-07	2.17191E-07	1.47439E-07
1.10000E+01	1.55262E-07	2.24202E-07	1.55209E-07
[illegible]	[illegible]	[illegible]	[illegible]
-1.30000E+01	1.87717E-07	2.39134E-07	1.71649E-07
1.40000E+01	2.06357E-07	2.47111E-07	1.80257E-07
1.50000E+01	2.26445E-07	2.55496E-07	1.89208E-07
1.60000E+01	2.49395E-07	2.64317E-07	1.98458E-07
1.70000E+01	2.74259E-07	2.73634E-07	2.08033E-07
-1.80000E+01	3.01729E-07	2.83515E-07	2.17931E-07
1.90000E+01	3.32154E-07	2.94038E-07	2.28160E-07
2.00000E+01	3.65953E-07	3.05302E-07	2.38724E-07
-2.10000E+01	4.03634E-07	3.17429E-07	2.49632E-07
2.20000E+01	4.45831E-07	3.30579E-07	2.60890E-07
2.30000E+01	4.93340E-07	3.44927E-07	2.72505E-07
-2.40000E+01	5.47189E-07	3.60766E-07	2.84483E-07
2.50000E+01	6.08747E-07	3.78423E-07	2.96831E-07
2.60000E+01	6.79887E-07	3.98415E-07	3.09557E-07
-2.70000E+01	7.63390E-07	4.21463E-07	3.22666E-07
2.80000E+01	8.63061E-07	4.48694E-07	3.36166E-07

CALCULATED MESHING ERRORS AND LOADS

J(C)	A(C)	TANG. ERROR	+2-PAIR	+1 PAIR	0 PAIR	-1-PAIR	WN	WT	QJD
			TANGENTIAL LOAD						
1.00000E+00	7.61598E-03	-1.81481E-03	0.	5.45328E+02	2.12287E+03	4.27729E+01	2.90384E+03	2.71097E+03	1.84067E-07
2.00000E+00	1.52320E-02	-1.80507E-03	0.	6.19441E+02	2.09153E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
-3.00000E+00	2.28479E-02	-1.78009E-03	0.	6.82616E+02	2.02836E+03	0.	2.90384E+03	-2.71097E+03	1.84067E-07
4.00000E+00	3.04639E-02	-1.75179E-03	0.	7.49044E+02	1.96193E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
5.00000E+00	3.80799E-02	-1.72243E-03	0.	8.18574E+02	1.89240E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
-6.00000E+00	4.56959E-02	-1.69432E-03	0.	8.91016E+02	1.81996E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
7.00000E+00	5.33119E-02	-1.66876E-03	0.	9.61133E+02	1.74484E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
8.00000E+00	6.09279E-02	-1.64110E-03	0.	1.04365E+03	1.66732E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
-9.00000E+00	6.85438E-02	-1.61319E-03	0.	1.12126E+03	1.58772E+03	0.	2.90384E+03	-2.71097E+03	1.84067E-07
1.00000E+01	7.61598E-02	-1.58491E-03	0.	1.20460E+03	1.50637E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
1.10000E+01	8.37758E-02	-1.55715E-03	0.	1.28732E+03	1.42365E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
1.20000E+01	9.13918E-02	-1.54002E-03	0.	1.38746E+03	1.32751E+03	0.	2.90384E+03	-2.71097E+03	1.84067E-07
1.30000E+01	9.90078E-02	-1.53417E-03	0.	1.49147E+03	1.21950E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
1.40000E+01	1.06624E-01	-1.52642E-03	0.	1.59966E+03	1.11132E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
1.50000E+01	1.14240E-01	-1.52153E-03	0.	1.70254E+03	1.00833E+03	0.	2.90384E+03	2.71097E+03	1.84067E-07
1.60000E+01	1.21856E-01	-1.52672E-03	0.	1.79126E+03	9.17718E+02	0.	2.90384E+03	2.71097E+03	1.84067E-07
1.70000E+01	1.29472E-01	-1.54263E-03	0.	1.87136E+03	8.39616E+02	0.	2.90384E+03	2.71097E+03	1.84067E-07
1.80000E+01	1.37088E-01	-1.55804E-03	0.	1.94766E+03	7.63308E+02	0.	2.90384E+03	2.71097E+03	1.84067E-07

1.90000E+01	1.44704E-01	-1.57316E-03	0.	2.02180E+03	6.89171E+02	0.	2.90384E+03	2.71097E+03	1.84067E-07
2.00000E+01	1.52320E-01	-1.58793E-03	0.	2.09342E+03	6.17557E+02	0.	2.90384E+03	2.71097E+03	1.84067E-07
2.10000E+01	1.59936E-01	-1.60250E-03	0.	2.16218E+03	5.48790E+02	0.	2.90384E+03	2.71097E+03	1.84067E-07
2.20000E+01	1.67552E-01	-1.46650E-03	2.94515E+02	2.05624E+03	3.60218E+02	0.	2.90384E+03	2.71097E+03	1.84067E-07
2.30000E+01	1.75148E-01	-1.46564E-03	3.50420E+02	2.09070E+03	2.69852E+02	0.	2.90384E+03	2.71097E+03	1.84067E-07
2.40000E+01	1.82784E-01	-1.44986E-03	4.08907E+02	2.11741E+03	1.84252E+02	0.	2.90384E+03	2.71097E+03	1.84067E-07
2.50000E+01	1.90400E-01	-1.43616E-03	4.69744E+02	2.13730E+03	1.03931E+02	0.	2.90384E+03	2.71097E+03	1.84067E-07

J(I)	A(J(I))	TANG. ERROR
1.00000E+00	7.61598E-03	-1.81481E-03
2.00000E+00	1.52320E-02	-1.80902E-03
3.00000E+00	2.28479E-02	-1.78006E-03
4.00000E+00	3.04639E-02	-1.75179E-03
5.00000E+00	3.80799E-02	-1.72393E-03
6.00000E+00	4.56959E-02	-1.69632E-03
7.00000E+00	5.33119E-02	-1.66876E-03
8.00000E+00	6.09279E-02	-1.64110E-03
9.00000E+00	6.85438E-02	-1.61319E-03
1.00000E+01	7.61598E-02	-1.59441E-03
1.10000E+01	8.37758E-02	-1.55615E-03
1.20000E+01	9.13918E-02	-1.54602E-03
1.30000E+01	9.90078E-02	-1.53417E-03
1.40000E+01	1.06624E-01	-1.52642E-03
1.50000E+01	1.14240E-01	-1.52153E-03
1.[illegible]E+01	1.21[illegible]E-01	-1.52[illegible]77E-03
1.70000E+01	1.29472E-01	-1.54263E-03
1.80000E+01	1.37088E-01	-1.55609E-03
1.90000E+01	1.44704E-01	-1.57316E-03
2.00000E+01	1.52320E-01	-1.58793E-03
2.10000E+01	1.59936E-01	-1.60256E-03
2.20000E+01	1.67552E-01	-1.46650E-03
2.30000E+01	1.75148E-01	-1.46564E-03
2.40000E+01	1.82784E-01	-1.44986E-03
2.50000E+01	1.90400E-01	-1.43616E-03

CALCULATED FOURIER COEFFICIENTS FOR ERRORS

I	A(I)	B(I)	C	KM
0	-3.1974709E-03	0.	3.1974709E-03	0.0000000
1	-3.7334873E-05	-9.3086172E-05	1.0029607E-04	1.0000000
2	4.2530368E-05	-5.8217736E-05	7.2098106E-05	2.0000000
3	2.5267277E-05	-5.3475548E-05	5.9144480E-05	3.0000000
4	1.3356568E-05	-4.1758555E-05	4.3842617E-05	4.0000000
5	9.0954477E-06	-2.4040746E-05	2.5704141E-05	5.0000000
6	9.4102963E-06	-1.4013834E-05	1.6880202E-05	6.0000000
7	1.5146503E-05	-4.5057904E-06	1.5802497E-05	7.0000000
8	2.0324879E-05	-4.4627445E-06	2.0852863E-05	8.0000000
9	2.1916959E-05	-6.7485406E-06	2.2932423E-05	9.0000000
10	1.8747237E-05	-9.0900698E-06	2.0834781E-05	10.0000000
11	1.4087511E-05	-8.0498529E-06	1.6225730E-05	11.0000000
12	1.0631619E-05	-3.2780210E-06	1.0553615E-05	12.0000000

AT1 angle of rotation, driving gear, between the position where the line of action intersects the involute at the start of the involute profile to the position where the line of action intersects the involute at the pitch point, rad.

AT2 angle of rotation, driven gear, rad.

Check for message to determine if part of the profile extends within the base circle. If so, program will continue and, where necessary, the root radius will be set equal to the base circle radius. In the tooth profile and the tooth deflection calculations, the original root circle radius will be used with the specified or calculated fillet radius. If the root circle lies inside the base circle by more than this fillet radius, a radial line is assumed to connect fillet and involute.

9. Driving Gear Data - J1, CJ1, AJ1, QJ1ABC, XJ1, YJ1, ZJ1, XME1(=X), YME1(=Y), J1, QJ1A, QJ1B, QJ1C

J1 Number of calculation points (see FI on Card 4 of the input data). Listed for values of (-M/2 + 1) to (M/2). (See description of Cards 8a, 8b, 8c and 8d.)

CJ1 Condition of engagement if equal to one and no engagement if equal to zero.

AJ1 Angle of rotation from the position of contact at the pitch point to the position of contact at the calculation point - negative for points inside the pitch circle, rad.

XJ1 and YJ1 Coordinates of the calculation point on the involute profile with the origin at the gear center and the X-axis as the centerline of the tooth. Given only for points at which contact takes place with the mating gear, in.

ZJ1 Tooth profile deviation, in.

XME1 and YME1 Coordinates of the point on the root circle midway between the tangent point of the fillet radius and the involute profile extended (and radial inside the base circle), in. This point is considered to be the end of the effective base of the tooth for deflection purposes.

QJ1A Elastic compliance normal to the profile of the gear tooth acting as a cantilever beam in bending only, in/lb.

QJ1B Elastic compliance of the gear tooth as a cantilever beam in shear only, in/lb.

QJ1C Elastic compliance of the gear tooth as a rigid member rotating in its supporting structure, in/lb.

QJ1ABC Combined compliance of QJ1A, QJ1B, QJ1C, QJ1ABC, in/lb.

10. Driven Gear Data - J2, BLANK, AJ2, QJ2ABC, XJ2, YJ2, ZJ2, XME2, YME2, J2, QJ2A, QJ2B, QJ2C

J2 Number of calculation points. For external gears, J2 is listed for values of (-M/2) to (M/2 - 1) and contact takes place between points where J1 = -J2. For internal gears, J2 is listed for values of (M/2) to (-M/2 + 1) and contact takes place between points where J1 = J2.

All other variables are similar to their counterparts for the driving gears.

11. Tooth meshing errors, loads and contact compliance - JC1, AJC1, TANG ERROR, TANG LOAD, WN, WT, QJD

JC1 Number of the calculation point on the driving gear tooth of the (0)th pair, starting with the first point after the pitch point and ending with the point corresponding to the pitch point of the next tooth ((1)st pair).

AJC1 Angle of rotation of the driving gear from the position with contact at the pitch point of the first mesh cycle to the position with contact at the calculation point, rad. The last angle in the first mesh cycle is the tooth spacing angle.

TANG ERROR Tooth meshing error or deviation from pure conjugate action, as a pitch-line linear measurement of the motion of the driven gear leading the driving gear, in. A negative value indicates that the driven gear is lagging the driving gear, as might be caused by deflection of the teeth.

TANG LOAD Load sharing among contacting tooth pairs is printed. The (+1)th pair is the pair following the (0)th pair, the (-1)th pair is the pair in front of the (0)th pair and the (+2)th pair follows the (+1)th pair, lb.

WN Total normal load transmitted by the teeth, lb.

WT Total tangential tooth load, lb.

QJD Contact or Hertzian compliance combined for both teeth at the contact point, in/lb.

If IHICR = 0, TANG LOAD is replaced by WTC and WTD as follows:

WTC Tangential load carried by the (0)th pair of teeth, lb.

WTD Tangential load carried by the (1)th pair of teeth, lb.

12. List of Tooth Meshing Error over NMC Mesh Cycles - JC1, AJC1, EJT

JC1 Number of the calculation point. The last value should be equal to (NMC x N_J).

AJC1 Same as AJC1 above.

EJT Same as TANG ERROR above.

13. Plot of Tooth Meshing Error.

Appears only if IPLT = 1 on input Card 2.

14. Fourier Coefficients - I, A, B, C, KM

I Order of the harmonic to which the coefficients apply. The zero order refers to the constant component.

A The Fourier coefficient of the cosine or real component for that harmonic of the meshing error, in. The value for I = 0 is twice the constant component or mean value of the meshing error.

B The Fourier coefficient of the sine or imaginary component for that harmonic of the meshing error, in.

C Square root of $(A^2 + B^2)$. Appears only if IFOUR = 1 on input Card 2.

KM Ratio of frequency to tooth mesh frequency.

The following output is concerned with the power spectral density function of the tooth meshing error. Appears only if ISPECT = 1 on input Card 2.

15. Mesh Frequency, cps.

16. Incidental Data - FC, FO, BE, H, TR, TMAX, SM

FC Cutoff frequency, cps.

FO Fundamental frequency, cps.

BE Equivalent resolution bandwidth, cps.

H Sampling interval, cps.

TR Total record time, sec.

TMAX Maximum displacement, sec.

SM Maximum lag number.

17. Power Spectral Density Function - K, FR, GK, GKK, KM

K Harmonic number, from 0 to SM.

FR Frequency, cps.

GK "Raw" power spectral density function, in^2.

GKK "Smooth" power spectral density function, in^2.

KM Ratio of frequency to tooth mesh frequency.

COMPUTER PROGRAM FOR PREDICTION OF GEAR MESH EXCITATION SPECTRA - GGEAR (R-67)

PROGRAM DESCRIPTION

Computer program GGEAR (R-67) is used for noise/vibration reduction analysis in order to calculate standard gear meshing errors and compliance. A listing of this program is provided in Appendix C. The output from this program is used in the finite element model of the internal components; in fact, R-67 is similar to program R-75 with the exception of the inclusion of high-contact-ratio capability, so the discussion of program R-75 is equally applicable here.

GGEAR is an outgrowth of GEARO (R-33). GEARO theory is thoroughly discussed in Reference 2 , and that theory shall not be repeated here. The differences between GEARO and GGEAR are outlined below, and GGEAR is thoroughly discussed in Reference 7.

Characteristics of vibration spectra induced by gear meshes in both single gear reductions and planetary gear reductions were first investigated. Methods were then developed to analyze the planet-pass induced vibrations which exist in normal planetary gear reduction systems. It was found that the planet-pass vibration sideband frequencies occurred both below and above the base signal at integer multiples of planet-pass frequency and that the sideband amplitudes could exceed those of the base signal. These sideband calculations were programmed into GEARO.

The effect of planet pair phasing on the vibration sideband spectra was analyzed for the CH-47 helicopter forward rotor drive transmission first-stage planetary reduction. GGEAR has been established for predicting vibration sidebands produced by variations in centerline distance, tooth transmitted force, and tooth support discontinuities for single gear mesh systems. The sidebands are normally found at mesh frequency harmonics plus and minus integer multiples of the frequency of variation of the gear parameters. The sideband amplitudes depend on the magnitude of variations of the gear parameters. The vibrations sideband spectra produced by spiral bevel gear shaft runout, externally imposed tooth mesh force variation, and a decrease in support stiffness over a number of consecutive ring gear teeth were obtained for gear meshes in the CH-47 helicopter forward rotor drive transmission. The GGEAR sideband analysis is useful both for designing low-vibration gear systems by properly controlling various gear parameters, and for identifying the existence of several types of gear problems such as gear runout, high dynamic tooth force, and tooth cracks.

COMPUTER PROGRAM R-67 INPUT FORMAT

A complete set of input data for R-67 comprises data of nine input cards. The input data instructions are given below and are for the most part self-explanatory. The cards are laid out in the format of an 80-column data card. A description of the variables is given and a sample case is provided in Figure 20.

Input Variables, Format, and Instructions

Card 1 Title, columns 2 through 72.

Card 2 Control numbers. Format (7I5)

a. NMC — Number of mesh cycles.

Place the last digit of this number in column 5.

b. INT — Identification as to whether this control card represents the last complete set of input data being submitted.
If more sets of input data follow, use 0.
If this is the last set, use 1.

Place this digit in column 10.

c. MN — Classification of the types of spur gears to be considered.
If both the driving and the driven gears are external gears, use 1.
If the driving gear is an external gear and if the driven gear is an internal (ring) gear, use 0.
(The program will not run properly if the internal gear is submitted as the driving gear.)

Place this digit in column 15.

d. MMM — Number of initial terms of the Fourier analysis for which coefficients will be printed, beyond the coefficient for the constant term. This number cannot exceed (FI x NMC + NMC/2), where FI is the input variable submitted on card 4.

Place the last digit of this number in column 20.

e. IPLT — Instruction as to whether the calculated tooth meshing error is to be plotted.
If tooth meshing error is to be plotted, use 1.
If plotting is to t bypassed, use 0.

Place this digit in column 25.

DATA PROCESSING WORKSHEET

NOTE In order to avoid misunderstanding in key punching, print these letters in the manner shown → i - I o - Ø s - S z - Z (See Office Instruction 24031)

1 of 2

GGEAR (R-67) INPUT SHEET

CARD 1 TITLE CARD COLUMNS 2 THRU 72

Ø SPIRAL BEVEL SUN GEAR MESH FØR R-67 INPUT TREGØLD APPRØX

CARD 2 CONTROL NUMBERS FØRMAT (7I5) RIGHT JUSTIFY

NMC	INT	MN	MMM	IPLT	IFOUR	ISPECT
1	1	1	7	1	1	1

CARD 3 GEAR DESIGN DATA FØRMAT(5E13.5) INCLUDE DECIMAL PØINT

FN1	FN2	PB1	RØ1	RØ2 OR RI2
0.45762 E02	0.17790 E03	0.42293 E01	0.48297 E01	0.17920 E02

CARD 4 GEAR DESIGN DATA FØRMAT(6E13.5) INCLUDE DECIMAL PØINT

RT1	RT2	RM1	RM2	FI	T1
0.44540 E01	0.17545 E02	0.44125 E01	0.17503 E02	0.12000 E02	0.38886 E00

CARD 5 GEAR DESIGN DATA FØRMAT (5E13.5) INCLUDE DECIMAL PØINT

T2	F1	F2	RF1	RF2
0.23977 E00	0.24142 E01	0.24142 E01	0.31161 E-01	0.31162 E-01

Figure 20. GGEAR (R-67) Input Sheet.

DATA PROCESSING WORKSHEET

NOTE

I - I   S - S
0 - Ø   Z - Z

2 .. 2

```
CARD 6 . . GEAR MATERIAL PRØPERTIES . . FØRMAT . (6E 13.5)
     YE1     |    YE2     |    GE1     |    GE2     |    PØS1    |    PØS2
3.0     E07|3.0     E07|1..1278  E07|1..1278  E07|0.33    E00|0.33    E00

CARD 7 . . CENTER DISTANCE AND TØØTH SPACING ERRØR FØRMAT (3E13.5)
     CL      |    VPT1    |    VPT2
0.22374  E02|5.0     E-04|5.0     E-04
            ADD 25 BLANK CARDS HERE
CARD 8  PØINT BY PØINT DATA . . FØRMAT (5E13.5)
     ZJ1     |    UJ1     |    ZJ2     |    UJ2     |    WT
                                                    |9.7     E03
            ADD 25 OF ABOVE CARD HERE
CARD 9 GEAR SPEED . . FØRMAT (I5, E13.5)
 NWS |  WS(RPM)
    1|.3      E03
```

Figure 20. Continued.

f. IFOUR Instruction as to whether Fourier analysis of the tooth meshing error is to be performed.
If it is to be performed, use 1.
If it is to be bypassed, use 0.

Place this digit in column 30.

g. ISPECT Instruction as to whether power spectral density function for the tooth meshing error is to be calculated.
If it is to be calculated, use 1.
If it is to be bypassed, use 0.

Card 3 Gear design data. Format (6E 13.5)

a. FN1 Number of teeth in the driving gear.

Use columns 1 through 13. (Do not omit decimal point.)

b. FN2 Number of teeth in the driven gear.

Use columns 14 through 26. (Do not omit decimal point.)

c. RB1 Base circle radius of driving gear, in.

Use columns 27 through 39.

d. RØ1 Radius to the outside diameter of the driving gear, in. This should be reduced by any radial loss in working surface at the tip of the teeth, as from tip rounding or chamfering.

Use columns 40 through 52.

e. RØ2 or RI2 Radius to the outside diameter of the driven gear, if external, and to the inside diameter, if internal, in. This should be corrected for any radial loss in working surface at the tip of the teeth, as from tip rounding or chamfering. In the case of an internal gear, this radius must be equal to or greater than the base circle radius. No check for this is provided.

Use columns 53 through 65.

Card 4 Gear design data, continued. Format (6E 13.5)

a. RT1 Radius to the beginning (near the base of the tooth) of the involute profile on the driving gear, in.

This is used in the program only in a design check as to whether adequate length of involute has been provided for contact on the teeth of the mating gear up to its tip. If this radius is not specified in the gear design data, this check may be bypassed by substituting the root circle radius.

Use columns 1 through 13.

b. RT2 Radius to the beginning (near the base of the tooth) of the involute profile on the driven gear, in. See above for substitute when not specified.

Use columns 14 through 26.

c. RM1 Radius to the root circle of the driving gear, in. If the radius submitted is smaller than the computed base circle radius, this is noted in the output, and the input value of root radius is used at some points in the program. If the root radius is sufficiently smaller than the base circle radius so that the root fillet center lies inside the base circle, the tooth outline between the base circle and the fillet is assumed to be a radial line by the program.

Use columns 27 through 39.

d. RM2 Radius to the root circle of the driven gear, in. For the case of an external gear, the same comments as above apply.

Use columns 40 through 52.

e. FI Number which indirectly establishes the number of calculation points. The number of these points will equal one plus twice the value of FI. The calculation points may be viewed as selected contact points on the true involute profile, extended where necessary. These contact points with the mating involute are associated with specific angular positions taken by the gear as it is rotated, where the angular positions correspond to uniform subdivisions of the tooth spacing angle. A greater number of these points will give more closely spaced point-by-point output data. A greater number will also give more accurate calculations of tooth deflections and

Fourier coefficients. A value of FI equal to 12 giving 25 calculation points has been found to be convenient.

Use columns 52 through 65. (Do not omit decimal point.)

f. T1 Circular tooth thickness at the pitch circle of the driving gear, in. The radius of the pitch circle is as defined in card 3. If not specified in the gear design data, it may be estimated as one-half of the difference between the actual circular pitch and the working backlash.

Use columns 66 through 78.

Card 5 Gear design data, continued. Format (5E 13.5)

a. T2 Circular tooth thickness at the pitch circle of the driven gear, in. The comments for T1 also apply here.

Use columns 1 through 13.

b. F1 Effective tooth face width of the driving gear, in. Where the face widths of the two gears are similar, use the actual face width without any reduction for normal end chamfering or rounding. Where one tooth is much wider, use as its effective face width an amount suitably larger than the narrower width to allow for the limited additional support that the greater width provides.

Use columns 14 through 26.

c. F2 Effective tooth face width of the driven gear, in. The comments for F1 also apply here.

Use columns 27 through 39.

d. RF1 Fillet radius on the driving gear, in.

Use columns 40 through 52.

e. RF2 Fillet radius on the driven gear, in.

Use columns 53 through 65.

Card 6 Gear material properties. Format (6E 13.5)

a. YE1 Young's modulus (in bending) for the material of the driving gear, lb/in.^{2}

Use columns 1 through 13.

b. YE2 Young's modulus (in bending) for the material of the driven gear, lb/in.^{2}

Use columns 14 through 26.

c. GE1 Shear modulus for the material of the driving gear, lb/in.^{2}

Use columns 27 through 39.

d. GE2 Shear modulus for the material of the driven gear, lb/in.^{2}

Use columns 40 through 52.

e. PØS1 Poisson's ratio for the material of the driving gear. Since this ratio is used only in the allowance for the "wide beam effect," it should be reduced for the cases where tooth face width is not much greater than tooth thickness, with a limiting value of zero when the teeth have a width smaller than the thickness.

Use columns 53 through 65.

f. PØS2 Poisson's ratio for the material of the driven gear. Comments for PØS1 also apply here.

Use columns 66 through 78.

There are NMC sets of the following Card 7 and Card 8. Each set supplies center distance, tooth spacing errors, tooth profile errors, tooth support compliances and tangential load for one mesh cycle.

Card 7 Center distance and tooth spacing error data. Format (3E 13.5)

a. CL Center distance, in.
This must be the actual center distance, including any substantial spreading under load.

Use columns 1 through 13.

b. VPT1 Tooth spacing error on the driving gear, in. This error is based on the distance between the pitch points of successive teeth, but the error is ad-

justed to apply to the direction of the line of action. This adjustment is accomplished by multiplying the pitch line error by the cosine of the pressure angle. The error is positive if the measured spacing is smaller than the desired spacing.

Use columns 14 through 26.

c. VPT2 Tooth spacing error on the driven gear, in. The comments under VPT1 also apply here.

Use columns 27 through 39.

Cards 8-1 to 8-2N Point-by-point data. Format (5E 13.5)

Total number of cards equal to twice the number of calculation points (N_J) between pitch points of adjacent teeth, or the same as two plus four times the value of F1 (see card 4). This specifies that cards must be introduced even if it is known that there is no contact at the particular calculation point or even if the tooth profile does not actually extend to the calculation point. As explained below, a blank card may be used for these points.

For the driving gear, the first card is for the calculation point located (N_J-1) points preceding the pitch point (or inside the pitch circle); the (N_J)th card is for the pitch point; the last or (2N_J)th card is for the calculation point located (N_J) points after the pitch point (or outside the pitch circle). The last point may also be described as the point of contact on one meshing tooth when the pitch point is the point of contact on the next meshing tooth.

For the driven gear which is an external gear, the first card is for the calculation point located (N_J) points before the pitch point (or inside the pitch circle); the (N_J+1)th card is for the pitch point; the last or (2N_J)th card is for the calculation point located (N_J-1) points after the pitch point (or outside the pitch circle). The point for the first card may also be described as the point of contact on one meshing tooth when the pitch point is the point of contact on the previous meshing tooth.

For the driven gear which is an internal gear, the first card is for the calculation point located (N_J) points following the pitch point (or outside the pitch circle); the (N_J+1)th card is for the pitch point; the last or ($2N_J$)th card is for the calculation point located (N_J-1) points before the pitch point (or inside the pitch circle). The point for the first card may also be described as the point of contact on the meshing tooth when the pitch point is the point of contact on the next meshing tooth.

a. ZJ1 Deviation of the point on the actual tooth profile on the driving gear from the true involute (as defined by the gear design data), in. This true involute is positioned relative to the actual profile so that its deviation at the pitch point is zero. Where the deviation represents material added to the true involute, it is positive; where it represents material subtracted, it is negative. The deviation is measured normal to the involute profile. If the profile does not extend to the particular calculation point or if it is known that the mating gear will not contact at this point, the deviation may be noted as zero.

Use columns 1 through 13.

b. UJ1 Tooth support compliance, or any compliance supplementary to the tooth compliance included in the analysis, on the driving gear, in./lb. This compliance is the deflection under unit load at the calculation point on the profile in the direction of the load (or normal to the profile). A uniform compliance for all calculation points, such as would result from a uniform gear shaft compliance, would not affect the final results as far as motion irregularities or load transfer is concerned; it would only increase the mean deviation in transmitted motion.

Use columns 14 through 26.

c. ZJ2 Deviation of the point on the actual tooth profile on the driven gear from the true involute, in. The comments under ZJ1 also apply here.

Use columns 27 through 39.

d. UJ2 Tooth support compliance, etc., on the driven gear, in./lb. The comments under UJ1 also apply here.

Use columns 40 through 52.

e. WT Total load, tangent to the pitch circle, transmitted by the gear teeth, lb.

In the first N card 8s, WT should be left blank. WT in the second N card 8s represents the loads at the N calculation points in one gear mesh.

Use columns 53 through 65.

Card 9 Gear Speed. Format (I5,E 13.5)

a. NWS Identification as to whether the input speed is the speed of driving or driven gear.
If driving gear speed is inputted, use 1.
If driven gear speed is inputted, use 2.

Place this digit in column 5.

b. WS Driving or driven gear speed, rpm.

Use columns 6 through 18.

PROGRAM OUTPUT

The output generated from the sample case defined in Figure 20 is shown in Figure 21. In addition to the calculated output data, the input data is also listed, thus producing a complete record of the computer run. Key output variables from R-67 are discussed briefly below.

Output Variables and Explanations

1. Title

2. Control numbers - NMC, INT, NMZ, MMM, as in input card 2.

3. Design - FI(=I), FN1(=N1), RB1, RO1, TR1, RM1
BLANK, FN2(=N2), BLANK, RO2 or RI2, RT2, RM2
T1, F1, RF1, YE1, GE1, PØS1
T2, F2, RF2, YE2, GE2, PØS2
all as in input cards 3 through 6.

Items 4-11 are printed for every mesh cycle, totally NMC cycles.

4. Mesh cycle identification and center distance.

5. Input listing of profile error and supplementary compliance - ZJ1, UJ1, ZJ2, UJ2 as in input card 8.

6. Pressure angle, degrees.

7. Incidental data — RP1, RB1, BA1, BR1, AT1
RP2, RB2, BA2, BR2, AT2

where:

where:	RP1	pitch circle radius of driving gear, in.
	RP2	pitch circle radius of driven gear, in.
	RB1	base circle radius of driving gear, in.
	RB2	base circle radius of driven gear, in.
	BA1	arc of approach of driving gear, rad.
	BA2	arc of approach of driven gear, rad (negative on internal gears).
	BR1	arc of recess of driven gear, rad.
	BR2	arc of recess of driving gear, rad (negative on internal gears).
	AT1	angle of rotation of driving gear from the position at which the line of action intersects the involute at the start of the involute profile to the position at which the line of action intersects the involute at the pitch point, rad.
	AT2	similar angle of rotation of driven gear, rad.

Check statement when part of the profile extends within the base circle.

Program will continue in any case, and, where necessary, the root radius will be set equal to the base circle radius. However, in calculating the tooth profile and the tooth deflections, the original root circle radius will be used with the specified fillet radius. If the root circle lies inside the base circle by more than this fillet radius, a radial line is assumed to connect fillet and involute.

8. Driving gear data — J1, CJ1, AJ1, QJ1ABC, XJ1, YJ1, XME1(=X), YME1(=Y), J1, QJ1A, QJ1B, QJ1C

where:	J1	identification number for calculation points (see under F1 of card 4 in the input data). Listed for values of (-2I) to (2I+1).
	CJ1	condition of engagement if equal to one and no engagement if equal to zero.
	AJ1	angle of rotation from the position of contact at the pitch point to the position of contact at the calculation point - negative for points inside the pitch circle, rad.
	XJ1 and YJ1	coordinates of the calculation point on the involute profile with the origin at the gear center and with the X-axis as the centerline of the tooth, given only for the points at which contact will take place with the mating gear, in.
	XME1 and YME1	coordinates of the point on the root circle midway between the tangent point of the fillet radius and the involute profile extended (and radial inside the base circle), in. This point is considered to be the end of the effective base of the tooth for deflection purposes.
	QJ1A	elastic compliance of the gear tooth acting as a cantilever beam in bending only, normal to the profile at the calculation point, in./lb.
	QJ1B	elastic compliance of the gear tooth as a cantilever beam in shear only; otherwise as above.
	QJ1C	elastic compliance of the gear tooth as a rigid member rotating in its supporting structure; otherwise as above.
	QJ1ABC	combined compliance of the three above, in./lb.

9. Driven gear data — J2, BLANK, AJ2, QJ2ABC, XJ2, YJ2, XME2, YME2, J2, QJ2A, QJ2B, QJ2C

where:	J2	identification number for the calculation points. For external gears, J2 is listed for values of (-2I-1) to (2I). For this case, contact takes place between points of the two gears for which J1 = -J2. For internal gears, J2 is listed for values of (2I+1) to (-2I). For this case, contact takes place between points of the two gears for which J1 = J2.

All other variables are similar to their counterparts for the driving gear.

10. Input tooth spacing error data — VPT1, VPT2
as in input card 7.

11. Tooth meshing errors, loads and contact compliance — JC1, AJC1, EJT, WTC, WTD, WN, WT, QJD

where: JC1 identification number for the calculation point on the first tooth of the driving gear, starting with the first point after the pitch point and ending with the point corresponding to the pitch point of the next tooth.

AJC1 angle of rotation of the driving gear from the position with contact at the pitch point of the first mesh cycle to the position with contact at the calculation point, rad. The last angle in the first mesh cycle is the tooth spacing angle.

EJT tooth meshing error or deviation from pure conjugate action, as a pitch-line linear measurement of the motion of the driven gear leading the driving gear, in. A negative value indicates that the driven gear is lagging the driving gear, as might be caused by deflection of the teeth.

WTC tangential load carried by the first pair of teeth, lb.

WTD tangential load carried by the second pair of teeth, lb.

WN total normal load transmitted by the teeth, lb.

WT input tangential tooth load, lb.

QJD contact or Hertzian compliance combined for both teeth at the contact point, in./lb.

12. List of tooth meshing error over NMC mesh cycles — JC1, AJC1, EJT

where: JC1 identification number for the calculation point. The last value should be equal to (NMCxN).

AJC1 same as AJC1 in item 11.

EJT same as EJT in item 11.

13. Plot of tooth meshing error.
    Appears only if IPLT in input card 2 is set to be 1.

14. Fourier coefficients — I, A, B, C, KM

where:

I	order of the harmonic to which the coefficients apply. The zero order refers to the constant component.
A	the Fourier coefficient of the consine or real component for that harmonic of the meshing error, in. The value for I = 0 is twice the constant component or mean value of the meshing error.
B	the Fourier coefficient of the sine or imaginary component for that harmonic of the meshing error, in.
C	square root of $(A^2 + B^2)$. Appears only if IFOUR in input card 2 is set to be 1.
KM	ratio of frequency to tooth mesh frequency.

The following output is concerned with the power spectral density function of the tooth meshing error. Appears only if ISPECT in input card 2 is set to be 1.

15. Mesh frequency, cps

16. Incidental data — FC, FO, BE, H, TR, TMAX, SM

where:

FC	cutoff frequency, cps.
FO	fundamental frequency, cps.
BE	equivalent resolution bandwidth, cps.
H	sampling interval, cps.
TR	total record time, sec.
TMAX	maximum displacement, sec.
SM	maximum lag number.

17. Power spectral density function — K, FR, GK, GKK, KM

where: K harmonic number, from 0 to SM.

FR frequency, cps.

GK "raw" power spectral density function, in.^{2}

GKK "smooth" power spectral density function, in.^{2}

KM ratio of frequency to tooth mesh frequency.

```
SPIRAL BEVEL SUN GEAR MESH FOR R-67 INPUT  TREGOLD APPROX

PNC335*COMPUTATIONS OF GEAR TOOTH MESHING ERRORS        3-22-1966

MESH CYCLES     INPUT,    DIFF.GEAR  N.OF HARMONIC
     1             1            1          7

     I              N1           RB1          RO1          RT1          RM1
0.12000E 02  0.45762E 02  0.42293E 01  0.48297E 01  0.44540E 01  0.44125E 01

                    N2                        RO2          RT2          RM2
0.0          0.17790E 03               0.17920E 02  0.17545E 02  0.17503E 02

     T1             F1           RF1     YOUNGS MOD-1 SHEAR MOD-1  POS.RATIO-1
0.38886E 00  0.24142E 01  0.31161E-01  0.30000E 08  0.11278E 08  0.33000E 00

     T2             F2           RF2     YOUNGS MOD-2 SHEAR MOD-2  POS.RATIO-2
0.23977E 00  0.24142E 01  0.31162E-01  0.30000E 08  0.11278E 08  0.33000E 00

CALCULATED DATA

MESH CYCLE NO.= 1,   CENTER DISTANCE= 0.223740E 02

INPUT LISTING OF PROFILE ERROR AND SUPPLEMENTATY COMPLIANCE
     Z1             U1           Z2           U2
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0
0.0          0.0          0.0          0.0

PRES.ANGLE(DEG)
0.22501E 02
```

Figure 21. R-67 Sample Run and Plot.

RP1	RB1	BA1	BB1	AT1
0.45778E 01	0.42293E 01	0.75007E-01	0.13720E 00	0.83960E-01

RP2	RB2	BA2	BB2	AT2
0.17796E 02	0.16441E 02	0.35291E-01	0.19294E-01	0.41744E-01

J1	CJ1	AJ1	QJ1ABC	XJ1	YJ1
-0.24000E 02	0.0	-0.13181E 00			
-0.23000E 02	0.0	-0.12632E 00			
-0.22000E 02	0.0	-0.12083E 00			
-0.21000E 02	0.0	-0.11533E 00			
-0.20000E 02	0.0	-0.10984E 00			
-0.19000E 02	0.0	-0.10435E 00			
-0.18000E 02	0.0	-0.98857E-01			
-0.17000E 02	0.0	-0.93365E-01			
-0.16000E 02	0.0	-0.87873E-01			
-0.15000E 02	0.0	-0.82381E-01			
-0.14000E 02	0.0	-0.76889E-01			
-0.13000E 02	0.10000E 01	-0.71397E-01	0.51419E-08	0.44650E 01	0.22985E 00
-0.12000E 02	0.10000E 01	-0.65905E-01	0.58141E-08	0.44727E 01	0.22762E 00
-0.11000E 02	0.10000E 01	-0.60413E-01	0.65209E-06	0.44806E 01	0.22531E 00
-0.10000E 02	0.10000E 01	-0.54921E-01	0.72650E-08	0.44885E 01	0.22292E 00
-0.90000E 01	0.10000E 01	-0.49428E-01	0.80487E-08	0.44965E 01	0.22045E 00
-0.80000E 01	0.10000E 01	-0.43936E-01	0.88748E-08	0.45047E 01	0.21790E 00
-0.70000E 01	0.10000E 01	-0.38444E-01	0.97463E-08	0.45129E 01	0.21526E 00
-0.60000E 01	0.10000E 01	-0.32952E-01	0.10566E-07	0.45213E 01	0.21254E 00
-0.50000E 01	0.10000E 01	-0.27460E-01	0.11638E-07	0.45298E 01	0.20974E 00
-0.40000E 01	0.10000E 01	-0.21968E-01	0.12666E-07	0.45383E 01	0.20684E 00
-0.30000E 01	0.10000E 01	-0.16476E-01	0.13753E-07	0.45470E 01	0.20386E 00
-0.20000E 01	0.10000E 01	-0.10984E-01	0.14904E-07	0.45558E 01	0.20079E 00
-0.10000E 01	0.10000E 01	-0.54921E-02	0.16124E-07	0.45647E 01	0.19763E 00
0.0	0.10000E 01	0.0	0.17417E-07	0.45737E 01	0.19437E 00
0.10000E 01	0.10000E 01	0.54921E-02	0.18788E-07	0.45828E 01	0.19102E 00
0.20000E 01	0.10000E 01	0.10984E-01	0.20243E-07	0.45919E 01	0.18758E 00
0.30000E 01	0.10000E 01	0.16476E-01	0.21789E-07	0.46012E 01	0.18404E 00
0.40000E 01	0.10000E 01	0.21968E-01	0.23431E-07	0.46106E 01	0.18041E 00
0.50000E 01	0.10000E 01	0.27460E-01	0.25179E-07	0.46201E 01	0.17667E 00
0.60000E [illegible]	0.10000E 01	0.32952E-01	0.27038E-07	0.46297E 01	[illegible]294E 00
0.70000E 01	0.10000E 01	0.38444E-01	0.29018E-07	0.46394E 01	0.16890E 00
0.80000E 01	0.10000E 01	0.43936E-01	0.31129E-07	0.46492E 01	0.16487E 00
0.90000E 01	0.10000E 01	0.49428E-01	0.33380E-07	0.46590E 01	0.16075E 00
0.10000E 02	0.10000E 01	0.54921E-01	0.35783E-07	0.46690E 01	0.15649E 00
0.11000E 02	0.10000E 01	0.60413E-01	0.38350E-07	0.46791E 01	0.15214E 00
0.12000E 02	0.10000E 01	0.65905E-01	0.41095E-07	0.46892E 01	0.14768E 00
0.13000E 02	0.10000E 01	0.71397E-01	0.44034E-07	0.46995E 01	0.14312E 00
0.14000E 02	0.10000E 01	0.76889E-01	0.47183E-07	0.47098E 01	0.13845E 00
0.15000E 02	0.10000E 01	0.82381E-01	0.50562E-07	0.47202E 01	0.13367E 00
0.16000E 02	0.10000E 01	0.87873E-01	0.54192E-07	0.47307E 01	0.12878E 00
0.17000E 02	0.10000E 01	0.93365E-01	0.58098E-07	0.47413E 01	0.12377E 00
0.18000E 02	0.10000E 01	0.98857E-01	0.62306E-07	0.47520E 01	0.11865E 00
0.19000E 02	0.10000E 01	0.10435E 00	0.66849E-07	0.47628E 01	0.11343E 00
0.20000E 02	0.10000E 01	0.10984E 00	0.71763E-07	0.47737E 01	0.10808E 00
0.21000E 02	0.10000E 01	0.11533E 00	0.77091E-07	0.47846E 01	0.10261E 00
0.22000E 02	0.10000E 01	0.12083E 00	0.82883E-07	0.47957E 01	0.97033E-01
0.23000E 02	0.10000E 01	0.12632E 00	0.89200E-07	0.48068E 01	0.91333E-01
0.24000E 02	0.10000E 01	0.13181E 00	0.96112E-07	0.48180E 01	0.85514E-01
0.25000E 02	0.0	0.13730E 00			

COOD. OF EFFECTIVE TOOTH PROFILE AT ROOT CIRCLE	X	Y
	0.440502E 01	0.256857E 00

J1	QJ1A	QJ1B	QJ1C
-0.13000E 02	0.11133E-09	0.50229E-08	0.76719E-11
-0.12000E 02	0.12235E-09	0.56915E-08	0.33034E-12
-0.11000E 02	0.13325E-09	0.63718E-08	0.15958E-10
-0.10000E 02	0.14489E-09	0.70645E-08	0.55556E-10
-0.90000E 01	0.15824E-09	0.77698E-08	0.12074E-09
-0.80000E 01	0.17441E-09	0.84876E-08	0.21273E-09
-0.70000E 01	0.19465E-09	0.92187E-08	0.33291E-09
-0.60000E 01	0.22037E-09	0.99633E-08	0.48264E-09
-0.50000E 01	0.25318E-09	0.10722E-07	0.66334E-09
-0.40000E 01	0.29483E-09	0.11495E-07	0.87643E-09
-0.30000E 01	0.34731E-09	0.12283E-07	0.11233E-08
-0.20000E 01	0.41283E-09	0.13086E-07	0.14055E-08
-0.10000E 01	0.49385E-09	0.13906E-07	0.17244E-08
0.0	0.59307E-09	0.14742E-07	0.20816E-08
0.10000E 01	0.71352E-09	0.15596E-07	0.24785E-08
0.20000E 01	0.85855E-09	0.16468E-07	0.29168E-08
0.30000E 01	0.10319E-08	0.17359E-07	0.33979E-08
0.40000E 01	0.12376E-08	0.18270E-07	0.39235E-08
0.50000E 01	0.14807E-08	0.19203E-07	0.44951E-08
0.60000E 01	0.17648E-08	0.20159E-07	0.51143E-08
0.70000E 01	0.20970E-08	0.21138E-07	0.57828E-08
0.80000E 01	0.24830E-08	0.22144E-07	0.65021E-08
0.90000E 01	0.29297E-08	0.23176E-07	0.72741E-08
0.10000E 02	0.34448E-08	0.24258E-07	0.81004E-08

0.11000E 02	0.40368E-08	0.25331E-07	0.89825E-08
0.12000E 02	0.47193E-08	0.26458E-07	0.99223E-08
0.13000E 02	0.54913E-08	0.27621E-07	0.10921E-07
0.14000E 02	0.63766E-08	0.28825E-07	0.11982E-07
0.15000E 02	0.73850E-08	0.30072E-07	0.13105E-07
0.16000E 02	0.85321E-08	0.31367E-07	0.14293E-07
0.17000E 02	0.98354E-08	0.32715E-07	0.15547E-07
0.18000E 02	0.11315E-07	0.34121E-07	0.16870E-07
0.19000E 02	0.12994E-07	0.35592E-07	0.18262E-07
0.20000E 02	0.14900E-07	0.37136E-07	0.19727E-07
0.21000E 02	0.17064E-07	0.38761E-07	0.21265E-07
0.22000E 02	0.19524E-07	0.40480E-07	0.22879E-07
0.23000E 02	0.22323E-07	0.42307E-07	0.24570E-07
0.24000E 02	0.25514E-07	0.44257E-07	0.26341E-07

J2	AJ2	QJ2ABC	XJ2	YJ2
-0.25000E 02	-0.35319E-01			
-0.24000E 02	-0.33906E-01	0.79709E-08	0.17589E 02	0.19982E 00
-0.23000E 02	-0.32493E-01	0.88824E-08	0.17598E 02	0.19676E 00
-0.22000E 02	-0.31080E-01	0.98540E-08	0.17606E 02	0.19368E 00
-0.21000E 02	-0.29668E-01	0.10886E-07	0.17614E 02	0.19057E 00
-0.20000E 02	-0.28255E-01	0.11981E-07	0.17623E 02	0.18745E 00
-0.19000E 02	-0.26842E-01	0.13145E-07	0.17631E 02	0.18429E 00
-0.18000E 02	-0.25429E-01	0.14384E-07	0.17639E 02	0.18112E 00
-0.17000E 02	-0.24017E-01	0.15697E-07	0.17648E 02	0.17792E 00
-0.16000E 02	-0.22604E-01	0.17093E-07	0.17656E 02	0.17470E 00
-0.15000E 02	-0.21191E-01	0.18575E-07	0.17665E 02	0.17146E 00
-0.14000E 02	-0.19778E-01	0.20145E-07	0.17673E 02	0.16819E 00
-0.13000E 02	-0.18366E-01	0.21916E-07	0.17682E 02	0.16489E 00
-0.12000E 02	-0.16953E-01	0.23587E-07	0.17691E 02	0.16158E 00
-0.11000E 02	-0.15540E-01	0.25460E-07	0.17699E 02	0.15824E 00
-0.10000E 02	-0.14127E-01	0.27462E-07	0.17708E 02	0.15487E 00
-0.90000E 01	-0.12715E-01	0.29583E-07	0.17717E 02	0.15148E 00
-0.80000E 01	-0.11302E-01	0.31532E-07	0.17725E 02	0.14807E 00
-0.70000E 01	-0.98892E-02	0.34221E-07	0.17734E 02	0.14463E 00
-0.60000E 01	-0.84765E-02	0.36756E-07	0.17743E 02	0.14117E 00
-0.50000E 01	-0.70637E-02	0.39449E-07	0.17752E 02	0.13769E 00
-0.40000E 01	-0.56510E-02	0.42307E-07	0.17760E 02	0.13418E 00
-0.30000E 01	-0.42382E-02	0.45351E-07	0.17769E 02	0.13064E 00
-0.20000E 01	-0.28255E-02	0.48584E-07	0.17778E 02	0.12708E 00
-0.10000E 01	-0.14127E-02	0.52019E-07	0.17787E 02	0.12349E 00
0.0	0.0	0.55682E-07	0.17796E 02	0.11988E 00
0.10000E 01	0.14127E-02	0.59579E-07	0.17805E 02	0.11625E 00
0.20000E 01	0.28255E-02	0.63726E-07	0.17814E 02	0.11259E 00
0.30000E 01	0.42382E-02	0.68158E-07	0.17823E 02	0.10891E 00
0.40000E 01	0.56510E-02	0.72875E-07	0.17832E 02	0.10520E 00
0.50000E 01	0.70637E-02	0.77933E-07	0.17841E 02	0.10146E 00
0.60000E 01	0.84765E-02	0.83319E-07	0.17850E 02	0.97700E-01
0.70000E 01	0.98892E-02	0.89096E-07	0.17859E 02	0.93913E-01
0.80000E 01	0.11302E-01	0.95293E-07	0.17868E 02	0.90103E-01
0.90000E 01	0.12715E-01	0.10194E-06	0.17877E 02	0.86265E-01
0.10000E 02	0.14127E-01	0.10909E-06	0.17886E 02	0.82403E-01
0.11000E 02	0.15540E-01	0.11679E-06	0.17895E 02	0.78514E-01
0.12000E 02	0.16953E-01	0.12510E-06	0.17905E 02	0.74599E-01
0.13000E 02	0.18366E-01	0.13409E-06	0.17914E 02	0.70659E-01
0.14000E 02	0.19778E-01			
0.15000E 02	0.21191E-01			
0.16000E 02	0.22604E-01			
0.17000E 02	0.24017E-01			
0.18000E 02	0.25429E-01			
0.19000E 02	0.26842E-01			
0.20000E 02	0.28255E-01			
0.21000E 02	0.29668E-01			
0.22000E 02	0.31080E-01			
0.23000E 02	0.32493E-01			
0.24000E 02	0.33906E-01			

COORD OF EFFECTIVE TOOTH PROFILE AT ROOT CIRCLE X Y
0.175013E 02 0.242182E 00

J2	QJ2A	QJ2B	QJ2C
-0.24000E 02	0.19481E-09	0.77244E-08	0.51658E-10
-0.23000E 02	0.21733E-09	0.85283E-08	0.13571E-09
-0.22000E 02	0.24299E-09	0.93478E-08	0.26321E-09
-0.21000E 02	0.27399E-09	0.10180E-07	0.43128E-09
-0.20000E 02	0.31286E-09	0.11027E-07	0.64123E-09
-0.19000E 02	0.36234E-09	0.11889E-07	0.89384E-09
-0.18000E 02	0.42546E-09	0.12769E-07	0.11902E-08
-0.17000E 02	0.50492E-09	0.13663E-07	0.15291E-08
-0.16000E 02	0.60448E-09	0.14576E-07	0.19129E-08
-0.15000E 02	0.72735E-09	0.15506E-07	0.23408E-08
-0.14000E 02	0.87727E-09	0.16455E-07	0.28130E-08
-0.13000E 02	0.10589E-08	0.17425E-07	0.33318E-08
-0.12000E 02	0.12761E-08	0.18415E-07	0.38959E-08
-0.11000E 02	0.15340E-08	0.19427E-07	0.45067E-08
-0.10000E 02	0.18371E-08	0.20461E-07	0.51636E-08
-0.90000E 01	0.21922E-08	0.21521E-07	0.58695E-08

-0.80000E 01	0.26041E-08	0.22606E-07	0.66224E-08
-0.70000E 01	0.30800E-08	0.23717E-07	0.74241E-08
-0.60000E 01	0.36260E-08	0.24856E-07	0.82739E-08
-0.50000E 01	0.42507E-08	0.26025E-07	0.91734E-08
-0.40000E 01	0.49612E-08	0.27224E-07	0.10122E-07
-0.30000E 01	0.57692E-08	0.28460E-07	0.11122E-07
-0.20000E 01	0.66820E-08	0.29729E-07	0.12173E-07
-0.10000E 01	0.77103E-08	0.31035E-07	0.13274E-07
0.0	0.88693E-08	0.32385E-07	0.14428E-07
0.10000E 01	0.10169E-07	0.33776E-07	0.15635E-07
0.20000E 01	0.11623E-07	0.35211E-07	0.16892E-07
0.30000E 01	0.13252E-07	0.36700E-07	0.18206E-07
0.40000E 01	0.15068E-07	0.38238E-07	0.19570E-07
0.50000E 01	0.17100E-07	0.39840E-07	0.20993E-07
0.60000E 01	0.19356E-07	0.41497E-07	0.22466E-07
0.70000E 01	0.21874E-07	0.43226E-07	0.23997E-07
0.80000E 01	0.24679E-07	0.45030E-07	0.25584E-07
0.90000E 01	0.27800E-07	0.46913E-07	0.27227E-07
0.10000E 02	0.31278E-07	0.48886E-07	0.28927E-07
0.11000E 02	0.35149E-07	0.50956E-07	0.30684E-07
0.12000E 02	0.39469E-07	0.53136E-07	0.32499E-07
0.13000E 02	0.44289E-07	0.55436E-07	0.34371E-07

INPUT DATA ON TOOTH SPACING ERROR

VPT1	VPT2
0.50000E-03	0.50000E-03

CALCULATED TOOTH MESHING ERRORS AND LOADS

JC1	AJC1	TANG. ERROR	WTC	WTD	WN	WT	QJD
0.10000E 01	0.54921E-02	-0.13169E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.20000E 01	0.10984E-01	-0.12944E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.30000E 01	0.16476E-01	-0.12752E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.40000E 01	0.21968E-01	-0.12593E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.50000E 01	0.27460E-01	-0.12467E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.60000E 01	0.32952E-01	-0.12372E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.70000E 01	0.38444E-01	-0.12309E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.80000E 01	0.43936E-01	-0.12277E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.90000E 01	0.49428E-01	-0.12278E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.10000E 02	0.54921E-01	-0.12310E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.11000E 02	0.60413E-01	-0.12375E-02	0.97000E 04	0.0	0.10499E 05	0.97000E 04	0.45073E-07
0.12000E 02	0.65905E-01	-0.78176E-03	0.60796E 04	0.36204E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.13000E 02	0.71397E-01	-0.77322E-03	0.59499E 04	0.37501E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.14000E	0.76889E-01	-0.76602E-03	0.58170E 04	0.38830E 04	0. 499E 05	0.97000E 04	0.45073E-07
0.15000E 02	0.82381E-01	-0.76014E-03	0.56809E 04	0.40191E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.16000E 02	0.87873E-01	-0.75545E-03	0.55416E 04	0.41584E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.17000E 02	0.93365E-01	-0.75195E-03	0.53995E 04	0.43005E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.18000E 02	0.98857E-01	-0.74957E-03	0.52544E 04	0.44456E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.19000E 02	0.10435E 00	-0.74830E-03	0.51069E 04	0.45931E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.20000E 02	0.10984E 00	-0.74815E-03	0.49573E 04	0.47427E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.21000E 02	0.11533E 00	-0.74903E-03	0.48052E 04	0.48948E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.22000E 02	0.12083E 00	-0.75105E-03	0.46517E 04	0.50483E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.23000E 02	0.12632E 00	-0.75414E-03	0.44965E 04	0.52035E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.24000E 02	0.13181E 00	-0.75842E-03	0.43401E 04	0.53599E 04	0.10499E 05	0.97000E 04	0.45073E-07
0.25000E 02	0.13730E 00	-0.13430E-02	0.0	0.97000E 04	0.10499E 05	0.97000E 04	0.45073E-07

*

JC1	AJC1	TANG. ERROR
0.10000E 01	0.54921E-02	-0.13169E-02
0.20000E 01	0.10984E-01	-0.12944E-02
0.30000E 01	0.16476E-01	-0.12752E-02
0.40000E 01	0.21968E-01	-0.12593E-02
0.50000E 01	0.27460E-01	-0.12467E-02
0.60000E 01	0.32952E-01	-0.12372E-02
0.70000E 01	0.38444E-01	-0.12309E-02
0.80000E 01	0.43936E-01	-0.12277E-02
0.90000E 01	0.49428E-01	-0.12278E-02
0.10000E 02	0.54921E-01	-0.12310E-02
0.11000E 02	0.60413E-01	-0.12375E-02
0.12000E 02	0.65905E-01	-0.78176E-03
0.13000E 02	0.71397E-01	-0.77322E-03
0.14000E 02	0.76889E-01	-0.76602E-03
0.15000E 02	0.82381E-01	-0.76014E-03
0.16000E 02	0.87873E-01	-0.75545E-03
0.17000E 02	0.93365E-01	-0.75195E-03
0.18000E 02	0.98857E-01	-0.74957E-03
0.19000E 02	0.10435E 00	-0.74830E-03
0.20000E 02	0.10984E 00	-0.74815E-03
0.21000E 02	0.11533E 00	-0.74903E-03
0.22000E 02	0.12083E 00	-0.75105E-03
0.23000E 02	0.12632E 00	-0.75414E-03
0.24000E 02	0.13181E 00	-0.75842E-03
0.25000E 02	0.13730E 00	-0.13430E-02

CALCULATED FOURIER COEFFICIENTS FOR ERRORS

I	AXI<	BXI<	C	KM
0	-0.1997923E-02	0.0	0.1997923E-02	0.0
1	-0.7584484E-04	-0.3095591E-03	0.3187149E-03	1.0000000
2	-0.4375001E-05	-0.2272174E-04	0.2313910E-04	2.0000000
3	-0.5979773E-04	-0.9571236E-04	0.1128566E-03	3.0000000
4	0.4939244E-05	-0.2052319E-04	0.2110918E-04	4.0000000
5	-0.5422770E-04	-0.4221746E-04	0.6872376E-04	5.0000000
6	0.1870019E-05	-0.2232479E-04	0.2240296E-04	6.0000000
7	-0.4757154E-04	-0.1511329E-04	0.4991455E-04	7.0000000

MESHING FREQUENCY IN CPS = 0.22881E 03

FC	FO	BE
0.28601E 04	0.22881E 03	0.57202E 04

H	TR	TMAX	SM
0.17482E	0.45704E-02	0.17482E-03	0.10000E 01

POWER SPECTRAL DENSITY FUNCTION

K	FR	GK	GKK	KM
0	0.0	0.75325E-10	0.22342E-10	0.0
1	0.28601E 04	-0.30641E-10	0.23034E 06	12.50000

CH-47 BEVEL SUN MESH USING TREGOLDS APP
ROX./TRUE INVOLUTE GEARS JS
TANG. FORCE = 9700.00 LB.

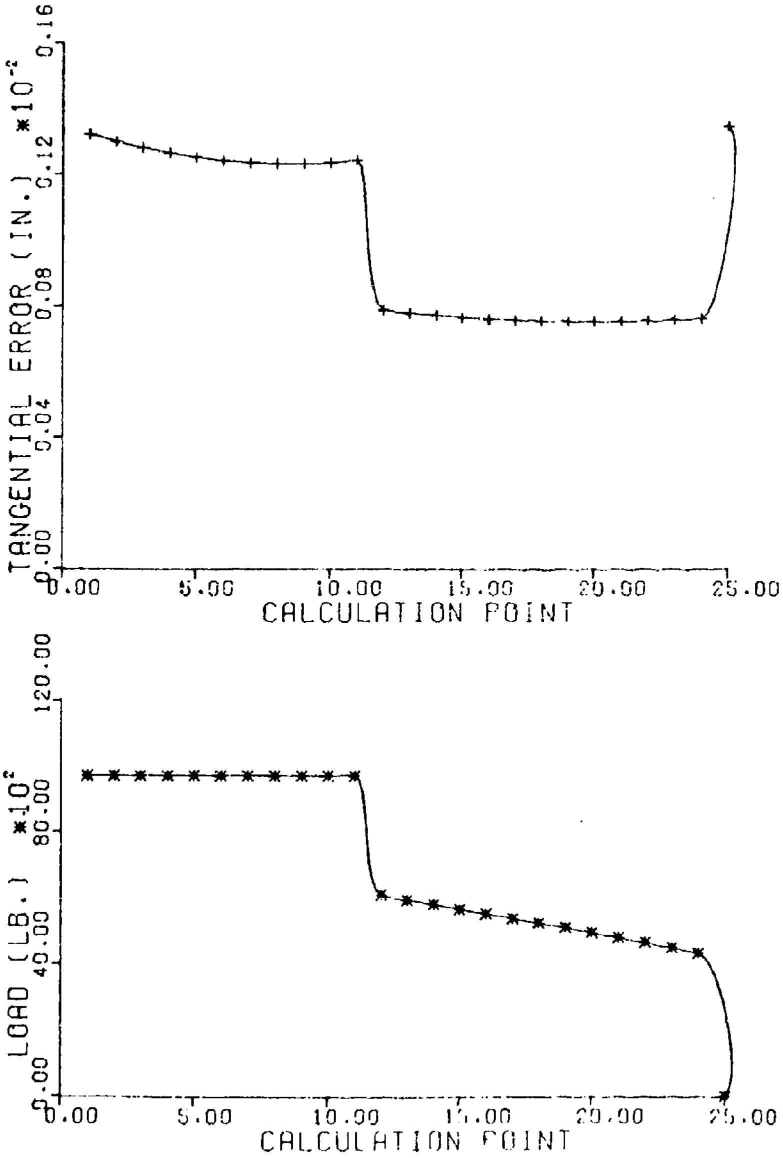


Figure 21. Continued.

CALCULATION OF GEAR MESH COMPLIANCE

In Reference 1, methodology is outlined in detail for approximately calculating the gear mesh compliance utilizing R-67 (GGEAR) output.

In Table 5, a worksheet is given for calculating the modified contact ratio using basic mesh data.

This modified contact ratio, along with GGEAR output data, is used to calculate the mesh compliance (Reference 1, page 156). An equivalent cross-sectional area must be calculated for input to the finite element model of the internal components. This calculation is shown in Table 6.

TABLE 5. CALCULATION OF MODIFIED CONTACT RATIO FOR CH-47 FORWARD ROTOR TRANSMISSION SPIRAL BEVEL GEARS

SYMBOL	DEFINITION	UNITS	PINION [114D1044(12)]	GEAR [114D1053(13)]
A_o	OUTER CONE DISTANCE	INCHES	7.134	7.134
a_o	LARGE END ADDENDUM	INCHES	.297	.146
D	LARGE END PITCH DIAMETER	INCHES	7.550	13.278
F'	FACE WIDTH	INCHES	2.188	2.188
P_d	LARGE END DIAMETRAL PITCH	—	3.841	3.841
Γ	PITCH ANGLE	DEGREES	31° 39'	67° 21'
Γ_o	FACE ANGLE	DEGREES	34° 44'	69° 13'
ϕ	NORMAL PRESSURE ANGLE	DEGREES	22° 30'	22° 30'
ψ	MEAN SPIRAL ANGLE	DEGREES	25° 00'	25° 00'
A	MEAN CONE DISTANCE $= A_o - (0.5)F'$	INCHES	6.100	6.100
σ	ADDENDUM ANGLE $= \Gamma_o - \Gamma$	DEGREES	3° 5'	1° 52'
a	MEAN ADDENDUM $= a_o - (0.5)F'(\mathrm{TAN}\sigma)$	INCHES	.238	.110
P	LARGE END TRANSVERSE CIRCULAR PITCH $= \pi/P_d$	—	.818	.818
P_n	MEAN NORMAL CIRCULAR PITCH $= (A/A_o)P(\mathrm{COS}\psi)$	—	.629	.629
P_2	FACTOR $= P_n/(\mathrm{COS}\phi)(\mathrm{COS}^2\psi + \mathrm{TAN}^2\phi)$	—	.685	.685
R	MEAN TRANSVERSE PITCH RADIUS $= (D/2\mathrm{COS}\Gamma)(A/A_o)$	INCHES	3.760	14.618
R_N	MEAN NORMAL PITCH RADIUS $= R/\mathrm{COS}^2\psi$	INCHES	4.578	17.796
R_{bN}	MEAN NORMAL BASE RADIUS $= R_N(\mathrm{COS}\phi)$	INCHES	4.229	16.442
R_{oN}	MEAN NORMAL OUTSIDE RADIUS $= R_N + a$	INCHES	4.816	17.907
$\Delta\rho$	PROPORTIONAL LENGTH OF ACTION $= \sqrt{R_{oN}^2 - R_{bN}^2} - R_N\mathrm{SIN}\phi$	INCHES	.551	.283
Z_N	LENGTH OF ACTION IN MEAN NORMAL SECTION $= \Delta\rho_P + \Delta\rho_G$	INCHES	.835	.835
K_2	FACTOR $= F'/A_o[2 - (F'/A_o)/2(1 - F'/A_o)]$	—	.371	.371
M_p	TRANSVERSE CONTACT RATIO $= Z_N/P_2$	—	1.219	1.219
M_F	FACE CONTACT RATIO $= [(K_2\mathrm{TAN}\psi - K_2^3/3\mathrm{TAN}^3\psi)A_oP_d]/\pi$	—	1.505	1.505
M_o	MODIFIED CONTACT RATIO $= \sqrt{M_p^2 + M_F^2}$	—	1.936	1.936

TABLE 6. GEAR TOOTH COMPLIANCE CALCULATION

$(q_{1-2})_{MIN}$ = (QJ1ABC + QJ2ABC) MIN + QJD

(QJ1ABC + QJ2ABC) MIN = 0.6248×10^{-7} IN./LB

QJD = 0.459319×10^{-7} IN./LB

$(q_{1-2})_{MIN} = 0.1084 \times 10^{-6}$ IN./LB = $0.1084\ \mu$ IN./LB

h = M-1 = 1.936-1 = 0.936

$$q_{R1} = \frac{(q_{1-2})\ MIN}{\cos^2 \psi \ \cos^2 \phi}$$

$\psi = 25^\circ \qquad \phi = 22.5^\circ$

$$q_{R1} = \frac{0.1084}{\cos^2 \psi \cos^2 \phi} = 0.1546\ \mu \text{ IN./LB}$$

$q_{1-2} = q_{R1}$ (1-0.3h) = 0.1546(1-0.3(0.936))

$q_{1-2} = 0.1112\ \mu$ IN./LB

$$k1\text{-}2 = \frac{1}{q_{1-2}} = 8.99 \times 10^6 \text{ LB/IN.}$$

$$\text{AREA} = A = \frac{Lk1\text{-}2}{E} = \frac{10(8.99 \times 10^6)}{30 \times 10^6} = \underline{\underline{2.998 \text{ IN.}^2}}$$

DYNAMIC GEAR TOOTH FORCE ANALYSES - TORRP (R-32)

PROGRAM DESCRIPTION

This program, which is based on a variation of the Holzer analysis, calculates the dynamic forces at a gear mesh. It can calculate system torsional resonances and dynamic tooth forces for a complex transmission system including planetary gears. The complete analysis is contained in Reference 2. The program source listing (Appendix D), input sheets, and a sample case have been included herein for completeness.

A gear system transmitting power is susceptible to torsional vibration since it possesses the necessary properties of rotational inertia, torsional elasticity, and a source of excitation. The inertia may be concentrated as in the body of a gear or distributed as in the shafting. Similarly, the elasticity (or compliance) may be concentrated as in a coupling or in the flexing gear teeth, or it may be distributed with the inertia in the shaft sections. As in any other torsionally vibrating system, the excitation may come from externally applied pulsating torques or from a fluctuating resistance to the steady rotation. However, in a geared system there is also an excitation due to displacement which comes from the imperfect transfer of motion between the meshing gears. Due to this excitation, the mating teeth at any point of excitation are subject to dynamic changes in relative motion, which can be achieved only if there is generated a dynamic force acting between the teeth to impose the necessary accelerations. This dynamic force, generated in response to the gear displacement excitation, can subject the gear teeth to greater loads than required for the steady transmission of power. In addition, it can become a factor in the generation of noise in the transmission system. The excitation, as well as the equivalent gear tooth compliance, is calculated from program GGEAR and used as input to TORRP.

Several features make the analysis sufficiently versatile to deal with a wide range of gear system designs. The analysis provides for branches in the vibrating system. It not only treats multiple cases of the simple gear set involving only one gear driving a second, but it can also treat multiple cases of one type of planetary gear set (the sun gear driving, the ring gear restrained, and the planet carrier transmitting the vibration to the balance of the system). Third, it includes the effects of externally applied damping such as might be developed at the bearings. The analysis presents the resulting dynamic gear tooth forces in a form giving their phase relationships as well as their magnitudes.

ORGANIZATION OF STATIONS WITHIN MAIN COMPUTER PROGRAM

The following provides a brief description of how to organize the data required to operate computer program - TORRP (R-32). Method of selecting and number stations will be presented.

Selection of Stations

Stations are established where one or more of the following descriptions apply:

a. The free end of any portion of the system, including the end and beginning of the main system and the end of any branch.

b. The point at which a concentrated inertia is acting. By concentrated inertia is meant an inertia which is not calculated by the program for given cylindrical dimensions.

c. The point at which a concentrated compliance is connected to the inertia which precedes or follows it. This inertia may be either concentrated or of the distributed type calculated by the program.

d. The point at which external constraints are applied to the system in the form of an elastic restraint or in the form of damping.

e. Each member of a simple gear set, but without any stations between these two.

f. The start and end of a planetary gear stage.

g. Where inertias and compliances are to be computed by the program, at any point where there are changes in any of the diameters used to define the cylindrical portions of the system. Any noncylindrical portion, such as a tapered portion, may be approximated by converting it into a series of stepped diameters.

Numbering of Stations

The stations are numbered in accordance with the following procedure. Refer also to Figure 22.

a. Select a "main system" which begins at a free end and trace a path through the system components until it reaches another free end. This path has the restriction that any branches which remain may not have further branching on themselves. Also, the path must enter each planetary stage through the sun gear. In Figure 22, which is a model of the closed-loop test stand and the

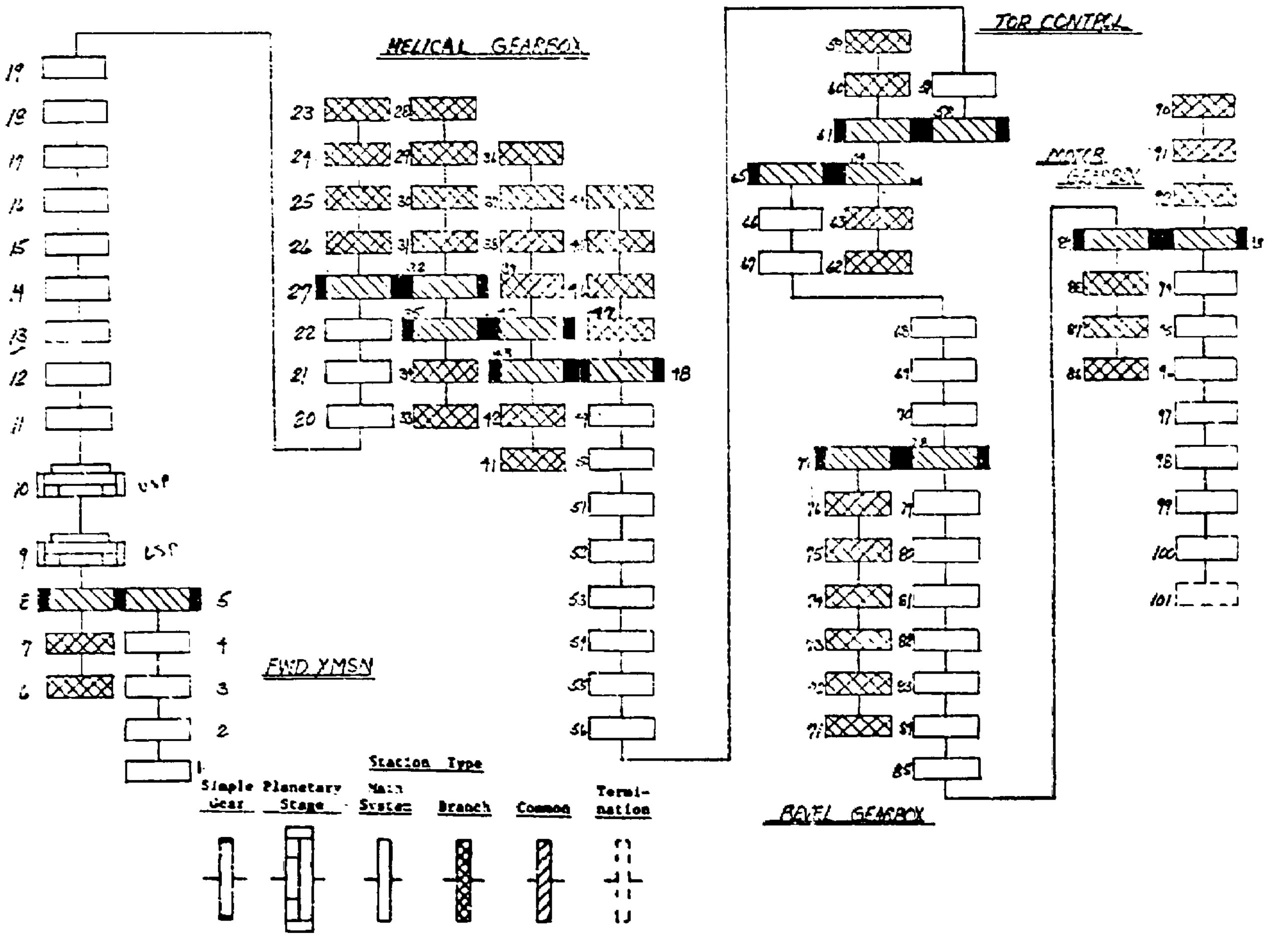


Figure 22. Schematic Diagram of Torsional System Model for CH-47 Forward Transmission and Closed-Loop Test Stand.

CH-47C forward transmission shown in Figure 23, the main system selected consists of the paths 1-5, 8-22, 27, 32, 35, 40, 43, 48-58, 61, 64-70, 78-85, 89, and 93-101.

b. Start numbering the stations beginning with 1 at the starting free end, continuing in sequence until a common station with a branch is reached. Do not give the common station the next number. Instead, go to the free, far end of the branch, continuing the numbering sequence at this far end and numbering along the branch until the common is reached. The common then receives the next number, and numbering is continued along the main system. The same procedure is followed whenever another branch is encountered. The end of the main system receives the highest station number.

PROGRAM (R-32) INPUT FORMAT

A set of input sheets required to program R-32 is shown in Figure 24. A complete set of input data comprises data of nine distinct categories. Within these categories, cards which convey specific kinds of information are referred to as card types. Depending on the complexity of the problem, the input data set may contain none, one or several cards of a given type. The categories are listed below:

- I Title Card
- II Control Number Card
- III Rotor Material Properties Card
- IV Rotor Data Card
- V External Constraints Cards
- VI Simple Gear Set Data Cards
- VII Planetary Gear Stage Data Cards
- VIII Branch Data Cards
- IX Gear Excitation Data Cards

A description of the input variables, format and instructions for each card is given below.

Card 1 Title, Format (72H). This card precedes each set of input data.

a. Printing instructions, column 1.
For printer to skip a line, use 0.
For printer to go to next sheet, use 1.

b. Title, columns 2 through 72.

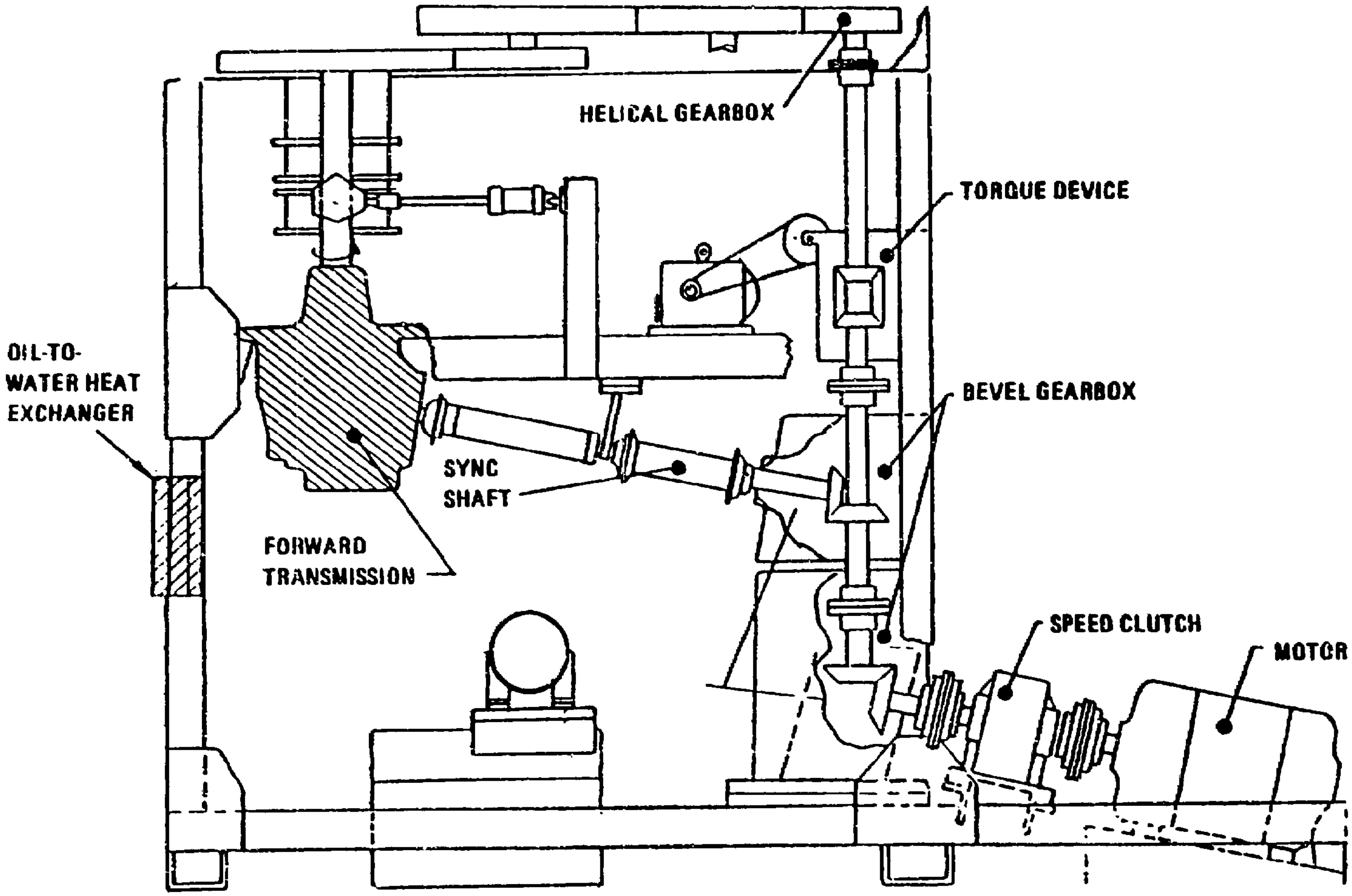


Figure 23. Schematic Diagram of Closed-Loop Test Stand.

First Item R-32 PAGE ____ OF ____

TORSIONAL RESPONSE PROGRAM

TORRP

INPUT SHEETS

ENGINEER ____________________ DATE ____________

PHONE __________

ALL NUMBERS RIGHT JUSTIFIED

TITLE - PLEASE PUNCH

CARD 1	0 (Col. 1)	2 ... 72

	NO. OF STATIONS	NO. OF EXT. CONST.	NO. OF BRANCHES	NO. OF EXCITN.	END 0=MORE 1=LAST	#SIMPLE GEAR SETS	#PLANET GEAR SETS	MDIAG
	1 5	6 10	11 15	16 20	21 25	26 30	31 35	36 40
CARD 2								

	MODULUS G LB/IN.^{2}	WEIGHT DENSITY LB/IN^3
	6 17	18 29
CARD 3		

Figure 24. TORRP (R-32) Input Sheet.

Second Item R-32 PAGE ____ OF ____

ALL NUMBERS RIGHT JUSTIFIED FACTOR DATA

Station 1 5	Concentrated Polar Inertia lb-in.^{2} 6 17	Length to Next Station in. 18 29	Stiffness O.D. in. 30 41	Mass O.D. in. 42 53	I.D. in. 54 65	Concentrated Compliance to Next Station rad/in-lb 66 77

Figure 24. Continued.

Third Item R-32 PAGE ____ OF ____

ALL NUMBERS RIGHT JUSTIFIED EXTERNAL CONSTRAINTS AND GEAR DATA

STATION NUMBER 1 5	6 17	18 29	30 41	42 53	54 65	66 77

BRANCH DEFINITION

END STATION NUMBER 1 5	COMMON STATION NUMBER 6 10	11 15

Figure 24. Continued.

Fourth Item R-32 PAGE ____ OF ____

ALL NUMBERS RIGHT JUSTIFIED GEAR EXCITATION DATA

Station	Frequency	Simple of Sun-Planet Excitation in.		Planet-Ring Excitation in.			
		Real	Imaginary	Real	Imaginary		
1 5	6 17	18 29	30 41	42 53	54 65	66 77	78 80

Figure 24. Continued.

Card 2 Control numbers. Format (7I5).

a. NS — Total number of stations. ($NS \leq 200$) Specification for selection of stations given in earlier section. (Place last digit in column 5.)

b. NB — Number of stations with external constraints, in the form of elastic restraint or damping, both with respect to ground. ($NB \leq NS$) (Place last digit in column 10.)

c. NBR — Number of branches, without including main system. ($NBR \leq 20$) (Place last digit in column 15.)

d. NMPG — Total number of cases of gear excitation. A change in station number, frequency, or magnitude of either component constitutes a separate case. In the planetary stations, one pair of sun and ring excitations of the same frequency constitutes one case. Individual solutions are found for each case, except in the planetary stages. When there are multiple excitations of the same frequency at different points in the system, the solutions for the individual excitations must be combined outside the program. (Place last digit in column 20.)

e. INP — Specifies if this is the last complete set of input data (column 25). If more sets of input data follow, use 0. If this is the last set, use 1.

f. NSRG — Number at simple gear sets. ($0 < NSRG \leq 20$) Each set consists of two gears. If an idler is used between two gears, the combination must be represented by two simple gear sets where the idler is replaced by two connected gears with no compliance between them and with a total inertia equal to that of the idler. Where one gear drives two or more gears, each leading off to separate branches, a similar conversion must be made. If one gear drives through multiple gears back into the main system, in a so-called star arrangement,

the only treatment possible in this program is to combine the multiple intermediate gears into one composite gear, assuming that the excitations, if any, are torsionally synchronous. (Place the last digit in column 30.)

g. NPLG Number of planetary gear stages (NPLG $\leq$ 2) (column 35).

Card 3 Rotor material properties. Format (5X, 2E12.4).

a. GM Shear modulus of elasticity, lb-in^2. May be zero only if all values of RL in the rotor data are zero (columns 6 through 17).

b. DENST Weight density, lb/in^3. May be zero only if all values of RL in the rotor data are zero (columns 18 through 29).

Card 4-1 to 4-NS Rotor data. Format (I5, 6E12.4).

a. NSTA Station number at or after which the rotor data apply. These must be given in numerical sequence with no omissions. (Place the last digit in column 5.)

b. RIP Moment of inertia concentrated at station, lb-in^2. This includes any inertia in the system which is not to be calculated by the program from dimensional data. At the stations for simple gear sets, list the inertias of both members in the rotor data. All planetary inertias are included only in the separate planetary data cards. The station which immediately follows the planetary may have its own inertia (columns 6 through 17).

c. RL Length of uniform cylindrical shaft section between this station and the adjoining, higher-numbered station, in. At the station for the first member of a simple gear set and at the station for a planetary gear stage, use 0.0. At the terminating or last station, use 0.0 or 1.0 (columns 18 through 29).

d. DST Outer diameter for stiffness calculation of the cylindrical shaft section, in. This diameter measures the section which transmits torque. If the actual shaft section is reduced as by a keyway, a diameter which approximates the reduced section should be used. At the station for the first member of a simple gear set and at the station for a planetary gear stage, use 0.0. At the terminating station, use 0.0 (columns 30 through 41).

e. DMS Outer diameter for mass calculation of the cylindrical shaft section, in. This diameter measures the section which contributes inertia. It may include any assembled sleeves or hubs which extend the full distance and which rotate with the shaft. Use 0.0 for the special stations as described under DST (columns 42 through 53).

f. DIN Inner diameter for both stiffness and mass calculation of the shaft section, in. If the shaft section is solid, use 0.0. Use 0.0 for the special stations as described under DST (columns 54 through 65).

g. CCOM Concentrated compliance acting between this station and the adjoining, higher-numbered station, rad/in-lb. This compliance is separate from that calculated by the program from the dimensional data, and it can be used only when there are no such dimensional data between the same two stations. Any value listed as a concentrated compliance will enter into the computation only if RL = 0 for the same station. Use 0.0 for the special stations as described under DST. Concentrated compliances associated with any of the gears are included only in the special data cards for the particular type of gear stage (columns 66 through 77).

Cards 5-1 to 5-NB External constraints. Format (I5, 2E 12.4) (These cards are omitted in NB = 0)

a. LB Station number at which the constraints are acting. These must be given in numerical sequence. (Place last digit in column 5.)

b. BK External, torsional, elastic restraint acting at the station, expressed as a torsional stiffness, in-lb/rad. In an actual system which has steady rotation, the stiffness must be 0 if the restraint is to ground. If in such a rotating system the restraint is to an "infinite" but rotating mass, the stiffness may have any finite value (columns 6 through 17).

c. BCB Coefficient of the external damping constraint acting at the station, in-lb-sec/rad (columns 18 through 29).

d. DMP Coefficient of interstation damping, in-lb-sec/rad (columns 30 through 41).

Cards 6-1 to 6-NSRG Simple gear set data. Format (I5, 4E 12.4) (At least one card must be submitted.)

a. LS Station number at which the first, or lower-numbered, member of the gear set is located. (The second member of the gear set is understood to be at the station LS+1, unless it serves as a common station to a branch. In this case, the second member has the number which completes the branch.) The simple gear set station numbers must be given in numerical sequence. (Place last digit in column 5.)

b. RP Pitch radius of the first member of the gear set at station LS, in. (columns 6 through 17). Negative.

c. RG Pitch radius of the second member of the gear set, in. (columns 18 through 29). Negative.

d. SG Combined linear compliance of the two gears, tangential to their pitch circles, in/lb (columns 30 through 41). (From GGEAR Output, A/2 W_t.)

Cards 7-1-A, B, C to 7-NPLG-A, B, C Planetary gear stage data. (These cards are omitted if NPLG = 0. When they are included, they must appear in sets of three, and each set must be arranged in the order given.)

First card of set - A Planetary geometry. Format (I5, 3E 12.4)

a. LEC — Station number at the start of the planetary stage. This location of the station is at the connecting point to the sun gear, but the station does not include the sun gear. The sun gear and other planetary components, including the planet carrier, lie between this station and the one following it. (Place the last digit in column 5.)

b. PN — Number of planet gears in the planetary stage (columns 6 through 17). Do not omit decimal point.

c. RS — Pitch radius of the sun gear, in. (columns 18 through 29).

d. RW — Pitch radius of the planet gear, in. The pitch radius of the ring gear will be calculated within the program by adding double the planet radius to the sun radius (columns 30 through 41).

Second card of set -B Planetary inertias. Format (I5, 5E 12.4)

a. IPL — Station number of the planetary stage, the same as LEC. (Place the last digit in column 5.)

b. PMS — Weight on one planet gear, lb. This includes all components which rotate with the planet - everything between bearing surface and gear teeth. One-half the weight of any rolling elements in the bearing should be included (columns 6 through 17).

c. PSP — Moment of inertia of the sun gear, lb-in^2. This includes all components between the point of connection to the outside system and the gear teeth (columns 18 through 29).

d. PIP Moment of inertia of each planet gear, lb-in^2. This includes all components used in computing the weight, PMS. (columns 30 through 41).

e. PRP Moment of inertia of the ring gear, lb-in^2. This includes all components between the gear teeth and the point of elastic connection to ground. If the connection to ground is rigid, with zero compliance, any finite inertia may be used for the ring gear (columns 42 through 53).

f. PCP Moment of inertia of the planet carrier, lb-in^2. This includes all components between the bearing surfaces in the planet gears and the point of connection to the outside system (columns 54 through 65).

Third card of set -C Planetary compliances. Format (I5, 5E 12.4)

a. IPL Station number of the planetary stage, a repetition of the value in the previous card. (Place the last digit in column 5.)

b. SS Combined linear compliance of each sun-planet gear mesh, tangential to its pitch circle, in-lb. If the sun gear construction is such that there is a significant compliance between the hub and the rim, the starting connection point to the sun gear should be defined as located at the rim. The hub would then be associated with the outside system as a separate station, with the structural compliance of the sun gear as a connecting concentrated compliance. In this case, the mesh compliance still appears under SS, but the sun gear inertia under PSP would be limited to that of the rim construction (columns 6 through 17) from GGEAR.

c. SR Combined linear compliance of each planet-ring gear mesh, tangential to its pitch circle, in-lb (columns 18 through 29) from GGEAR.

d. SW Linear compliance of the planet support in the planet carrier, tangential to the path of planet centers, in-lb. This compliance is the combination of the compliance of the planet bearing and the compliance of any portion of the carrier which will deflect with the individual planet. If the carrier construction is such that there is a compliance between a hub and a rim-type member which supports all the planets collectively, this structural compliance may be combined with the others to give a total planet carrier compliance. Alternatively, the system may be changed so that the end of the planetary stage, that is, its connection to the external system, is taken at the rim-type member. In this case, the hub becomes associated with the outside system as a separate station. The compliance between hub and rim then appears as a separate concentrated compliance between the station at the close of the planetary stage and the new station for the hub. With this change, the compliance used under SW is the combination of only the first two compliances mentioned above; namely, those associated with the individual planet (columns 30 through 41).

e. Blank Columns 42 through 53 are not read in this main computer program.

f. ST Angular compliance of the support between the ring and ground, rad/in-lb. If the ring is rigidly connected to ground, set this compliance equal to zero (columns 54 through 65).

Cards 8-1 to 8-BR Branch data. Format (2I5) These cards are omitted if BR = 0.)

a. LBR Number of the first or free end station of the branch. (Place the last digit in column 5.)

b. LBS Number of the common station of the branch at which it is connected to the main system. (Place the last digit in column 10.)

Cards 9-1 to 9-NMPG Gear excitation data. Format (I5, 6E 12.4) (These cards may be submitted in any order.)

a.	IT	Station number which identifies the gears at which the excitation is introduced. For simple gear sets, the number of the first member is used, as for LS in card 6. For planetary gear stages, the station number at the start of the planetary is used, as for LEC in card 7-A. (Place the last digit in column 5.)
b.	FFQ (TE1)	Frequency of the excitation, cps. (TE1 is used for temporary storage.) (columns 6 through 17.)
c.	AXY or AXY1 (TE2)	The real or cosine component of the linear excitation in the simple gear set (AXY) or in the sun-planet mesh of the planetary gear stage (AXY1), in. This excitation is introduced at the gear mesh tangential to the pitch circles. (TE2 is used for temporary storage.) (columns 18 through 29.)
d.	BXY or BXY (TE3)	The imaginary or sine component of the linear excitation described above, in. (columns 30 through 41).
e.	AXY2 (TE4)	The real or cosine component of the linear excitation.in the planet-ring mesh of the planetary gear stage, in. This excitation is introduced at the gear mesh tangential to the pitch circles. On a card with excitation for a simple gear set, this field is left blank. (TE4 is used for temporary storage.) (columns 42 through 53)
f.	BXY2 (TE5)	The imaginary or sine component of the linear excitation described just above, in. (columns 54 through 65).

PROGRAM OUTPUT

The output data generated from program R-32 is shown in Figure 25, along with the input data previously described. This procedure provides a complete record of the computer run. Key output variables from R-32 are discussed briefly below.

Tabulation of Input Data

1. Title - as in input card 1.
2. Control numbers - NS, NB, NBR, NMPG, INP, NSRG, NPLG, as in input card 2.
3. Rotor material properties - GM, DENST, as in input card 3.
4. Rotor data - NSTA, RIP, RL, DST, DMS, DIN, CCOM, as in input card 4.
5. External constraint data - LB, BK, BCB, as in input card 5.
6. Simple gear set data - LS, RP, RG, SG, as in input card 6.
7. Planetary gear stage data - LEC, PN, RS, RW, IPL, PMS, PSP, PID, PRP, PCP, SS, SR, SW, Blank, ST, as in input card 7.
8. Branch data - LBR, LBS, as in input card 8.
9. Excitation data - IT, FFO, AXY or AXX1, BXY or BXY1, AXY2, BXY2, as in input card 9. Each set of excitation data is given separately followed by its own calculated response.

Calculated Data

Computed response at each simple gear set - LS, TTFR, TTFE

where LS - Station number identifying the simple gear set.

TTFR - Real or cosine component of the dynamic tangential tooth force developed at the gear mesh, lb.

TTFE - Imaginary or sine component of the same force, lb.

```
//BVE053P    .B TNE,V746CR3200,000000,2054,3-01,,95,9999,,,,601,1356
OCH-47 SPIRAL BEVEL GEAR MESH R32 BADGLEY
   53    0   05    1    1    2    2
          11.32E6           .281
    1  652.9        1.0            4.12          4.12          0.0          0.0
    2    0.0        1.56           3.54          5.512         0.0          0.0
    3    0.0        4.25           3.348         5.216         0.0          0.0
    4   46.3        3.828          3.0           3.0           0.0          0.0
    5   101.658    25.86           4.346         4.337        -4.25         0.0
    6   46.388     26.68           4.346         4.337         4.25         0.0
    7   89.388      1.15           2.10          2.1           0.           0.0
    8    0.0        2.04           2.10          2.1           0.           0.0
    9    0.0        0.81           2.10          2.1           0.           0.0
   10    0.0        2.20           4.32          4.75          3.6          0.0
   11    0.0        0.97           4.32          4.65          3.6          0.0
   12    0.0        1.50           4.32          4.95          3.6          0.0
   13    0.0        3.28           4.50          5.3           3.6          0.0
   14    0.0        0.75           7.00          7.0           0.           0.0
   15    0.3653     2.2            3.16          3.16          2.9          0.0
   16   72.3        0.             0.00          0.            0.           0.0
   17   -0.216      0.95           0.86          0.86          0.52         0.0
   18    0.1293     1.00           2.50          2.50          2.14         0.0
   19    0.1293     9.00           0.64          0.64          0.50         0.0
   20    0.174      3.4            0.64          0.64          0.50         0.0
   21    4.199      4.20           3.00          3.3           2.50         0.0
   22   26.20       0.93           3.60          3.60          0.           0.0
   23    0.         2.50           4.10          4.10          0.           0.0
   24    0.         0.47           4.30          4.30          0.           0.0
   25 568.0         1.0            6.00          6.0           0.           0.0
   26    0.         2.3            6.00          6.0           0.           0.0
   27    0.0        0.0            0.00          0.0           0.0          0.0
   28    0.0        0.0            0.00          0.0           0.0          0.0
   29    0.0        7.08           6.90          6.9           6.1          0.0
   30    0.0       11.9            5.86          6.0           4.7          0.0
   31    0.0       11.5            5.30          5.3           3.4          0.0
   32 2148.743      7.955          5.625        11.22          3.25         0.0
   33 2656.536      1.235         11.251        13.109         0.0          0.0
   34    0.0        2.0           10.919        10.919         0.0          0.0
   35    0.0       10.88           8.178         9.0           0.0          0.0
   36 51673.308    21.406         18.25         18.25         16.624        0.0
   37 51673.308    15.75           8.178         9.00          0.0          0.0
   38   0.0         3.0            8.25          8.25          0.0          0.0
   39   0.0         2.75           9.333         9.333         0.0          0.0
   40   0.0         4.62           9.752        12.7967        0.0          0.0
   41   0.0         1.53           6.985         6.985         0.0          0.0
   42   0.0         2.03           7.483        10.433         0.0          0.0
   43   0.0         0.69           8.25          8.25          0.0          0.0
   44   0.0         1.0           11.50         11.5           0.0          0.0
   45    4.8056E6   0.0            0.00          0.0           0.0          0.0
   46    0.0        1.4            5.834         5.834         0.625        0.0
   47    0.0        2.6            6.30          9.3425        0.0          0.0
   48    0.0        0.69           7.04          7.04          0.0          0.0
   49    0.0        0.87           9.00          9.0           0.0          0.
   50 62347.3      11.76           8.02         20.0           0.0          0.
   51    0.0        1.44           5.89          6.758         0.0          0.
   52    0.0        6.62           7.00          9.218         0.0          0.
   53  .72786 E6    0.0            0.00          0.0           0.0          0.
   16    3.201     -5.629          0.1126E-06
   45   35.227    -10.0155         0.0153E-06
   27    4.0        2.8           03.9
   27   14.3       34.7          117.8          2660.0         1218.
   27  0.1717E-06  0.1866E-06     00.10 E-05                   0.0
   28    6.0        4.0           03.3
   28      13.75        254.          110.         4780.          4050.
   28      .874E-7     .8915E-7       .5E-6                        0.0
```

Figure 25. Sample Output for R-32 Torsional Response.

```
17    25
41    45
46    50
51    53
16       117.66 16.35070E-6          0.
/*
//
```

CH-47 SPIRAL BEVEL GEAR MESH R32 BADGLEY

TORSIONAL RESPONSE OF THE SYSTEM WITH GEARS,EPICYCLIC GEARS AND BRANCHES PN408.

STATIONS	BRG+EXT.CON.	BRANCHES	NO.OF FRQ	INP SETS	SR GEARS	PL GEARS
53	0	5	1	1	2	2

SHEAR MOD.	WT. DENSITY
LBS/IN**2	LBS/IN**3
0.115200 08	0.281000 00

ROTOR DATA

STA NO.	MOM.INERT. LBS-IN**2	LENGTH IN	STIFFN.DIA IN	MASS DIA IN	INNER DIA IN	CONC. COMPL. RAD/IN-LB
1	0.652900 03	0.100000 01	0.412000 01	0.412000 01	0.0	0.0
2	0.0	0.156000 01	0.354000 01	0.551200 01	0.0	0.0
3	0.0	0.425000 01	0.334800 01	0.521600 01	0.0	0.0
4	0.463000 02	0.382800 01	0.300000 01	0.300000 01	0.0	0.0
5	0.101660 03	0.259600 02	0.434600 01	0.433700 01	0.425000 01	0.0
6	0.463880 02	0.266800 02	0.434600 01	0.433700 01	0.425000 01	0.0
7	0.893880 02	0.115000 01	0.210000 01	0.210000 01	0.0	0.0
8	0.0	0.204000 01	0.210000 01	0.210000 01	0.0	0.0
9	0.0	0.810000 00	0.210000 01	0.210000 01	0.0	0.0
10	0.0	0.220000 01	0.432000 01	0.475000 01	0.360000 01	0.0
11	0.0	0.970000 00	0.432000 01	0.465000 01	0.360000 01	0.0
12	0.0	0.158900 01	0.432000 01	0.495000 01	0.360000 01	0.0
13	0.0	0.328000 01	0.450000 01	0.530000 01	0.360000 01	0.0
14	0.0	0.750000 00	0.700000 01	0.700000 01	0.0	0.0
15	0.365300 00	0.220000 01	0.316000 01	0.316000 01	0.290000 01	0.0
16	0.723000 02	0.0	0.0	0.0	0.0	0.0
17	0.216900 00	0.950000 00	0.860000 00	0.860000 00	0.520000 00	0.0
18	0.129300 00	0.100000 01	0.250000 01	0.250000 01	0.214000 01	0.0
19	0.129300 00	0.900000 01	0.640000 00	0.640000 00	0.500000 00	0.0
20	0.174000 00	0.340000 01	0.640000 00	0.640000 00	0.500000 00	0.0
21	0.419900 01	0.420000 01	0.300000 01	0.330000 01	0.250000 01	0.0
22	0.262000 02	0.330000 00	0.360000 01	0.360000 01	0.0	0.0
23	0.0	0.250000 01	0.410000 01	0.410000 01	0.0	0.0
24	0.0	0.470000 00	0.430000 01	0.430000 01	0.0	0.0
25	0.568000 03	0.100000 01	0.600000 01	0.600000 01	0.0	0.0
26	0.0	0.230000 01	0.600000 01	0.6000 0 01	0.0	0.0
27	0.0	0.0	0.0	0.0	0.0	0.0
28	0.0	0.0	0.0	0.0	0.0	0.0
29	0.0	0.708000 01	0.690000 01	0.690000 01	0.610000 01	0.0
30	0.0	0.119000 02	0.586000 01	0.600000 01	0.470000 01	0.0
31	0.0	0.115000 02	0.530000 01	0.530000 01	0.340000 01	0.0
32	0.214870 04	0.795500 01	0.562500 01	0.112200 02	0.325000 01	0.0
33	0.265650 04	0.123500 01	0.112510 02	0.131090 02	0.0	0.0
34	0.0	0.200000 01	0.109190 02	0.109190 02	0.0	0.0
35	0.0	0.108800 02	0.817800 01	0.900000 01	0.0	0.0
36	0.516730 05	0.214060 02	0.182500 02	0.182500 02	0.166240 02	0.0
37	0.516730 05	0.157500 02	0.817800 01	0.900000 01	0.0	0.0
38	0.0	0.300000 01	0.825000 01	0.825000 01	0.0	0.0
39	0.0	0.275000 01	0.933300 01	0.933300 01	0.0	0.0
40	0.0	0.462000 01	0.975200 01	0.127970 02	0.0	0.0
41	0.0	0.153000 01	0.698500 01	0.698500 01	0.0	0.0
42	0.0	0.203000 01	0.748300 01	0.104330 02	0.0	0.0
43	0.0	0.690000 00	0.825000 01	0.825000 01	0.0	0.0
44	0.0	0.100000 01	0.115000 02	0.115000 02	0.0	0.0
45	0.488560 07	0.0	0.0	0.0	0.0	0.0
46	0.0	0.140000 01	0.583400 01	0.583400 01	0.625000 00	0.0
47	0.0	0.260000 01	0.630000 01	0.934250 01	0.0	0.0
48	0.0	0.690000 00	0.704000 01	0.704000 01	0.0	0.0

```
 49  0.0             0.87000D 00 0.90000D 01 0.900000 01 0.0           0.0
 50  0.52347D 05 0.11750D 02 0.80200D 01 0.20000D 02 0.0           0.0
 51  0.0             0.14400D 01 0.58900D 01 0.67580D 01 0.0           0.0
 52  0.0             0.66200D 01 0.70000D 01 0.92180D 01 0.0           0.0
 53  0.72786D 06 0.0             0.0             0.0             0.0           0.0

        SINGLE REDUCTION GEAR DATA

STA   GEAR SET RADII (IN.)  COMBINED TANGENTIAL COMPLIANCES

      FIRST GEAR  SECOND GEAR   (IN/LB)
 16  0.32010D 01-0.56290D 01 0.11260D-06
 45  0.35227D 02-0.10016D 02 0.15300D-07

             PLANETARY SET DATA

STA    NO. OF         SUN         PLANET
NO.    PLANETS     RADIUS(IN)  RADIUS(IN)
 27  0.40000D 01 0.28000D 01 0.39000D 01

STA  WEIGHT(LB)     POLAR MASS MOMENTS OF INERTIA (LBS-IN**2)
NO.    PLANET          SUN         PLANET         RING        CARRIER
 27  0.14380D 02 0.74700D 02 0.11760D 03 0.26600D 04 0.12180D 04

STA       COMPLIANCE- LINEAR(IN./LB)                      COM-ANG(RAD/LB)
NO.  SUN-PLANET  PLANET-RING  PLAN.-CAR.                   RING-GROUND
 27   0.1717D-06   0.1966D-05   0.1000D-05                      0.0

STA    NO. OF         SUN         PLANET
NO.    PLANETS     RADIUS(IN)  RADIUS(IN)
 28  0.60000D 01 0.40000D 01 0.33000D 01

STA  WEIGHT(LB)     POLAR MASS MOMENTS OF INERTIA (LBS-IN**2)
NO.    PLANET          SUN         PLANET         RING        CARRIER
 28  0.13750D 02 0.25480D 03 0.11000D 03 0.47800D 04 0.40590D 04

STA       COMPLIANCE- LINEAR(IN./LB)                      COM-ANG(RAD/LB)
NO.  SUN-PLANET  PLANET-RING  PLAN.-CAR.                   RING-GROUND
 28   0.8740D-07   0.8915D-07   0.5000D-06                      0.0

  BRANCHES
   FAR END      COMMON
      15          16
      17          25
      41          45
      46          50
      51          53

                SINGLE REDUCTION GEAR- LINEAR EXCITATION (IN.)
       STA  FREQUENCY     REAL        IMAGINARY
       16  0.11766D 03 0.16351D-04 0.0

         COMPUTED RESPONSE AT SIMPLE GEAR SET

       STA  TANGENTIAL TOOTH FORCE (LBS)
               REAL       IMAGINARY
       16  0.10786D 02 0.0
       45  0.28385D 02 0.0

         COMPUTED RESPONSE AT PLANETARY GEAR SETS

           TANGENTIAL TOOTH FORCE AT EACH PLANET (LBS)
       STA         SUN-PLANET                    RING-PLANET
       NO.       REAL         IMAGINARY        REAL          IMAGINARY
       27  0.12708D 01 0.0              0.63120D 00 0.0
       28  0.36521D-01 0.0             -0.21607D 00 0.0
```

Computer response at each planetary gear stage - LEC, C1, C2, C3, C4

where LEC	-	Station number identifying the planetary gear stage.
C1	-	Real component of the dynamic tangential tooth force developed at the sun-planet gear mesh, lb.
C2	-	Imaginary component of the same force, lb.
C3	-	Real components of the dynamic tangential tooth force developed at the planet-ring gear mesh, lb.
C4	-	Imaginary component of the same force, lb.

This program R-32 is written in FORTRAN II - extended and may be compiled with FORTRAN IV. In the source deck listing shown in Appendix D, the READ and WRITE statements are written with the variable NR = 5 to specify the standard reading unit and the variable NW = 6 to specify the standard writing unit. To recompile the program for any nonstandard computer, introduce the required unit numbers, making the necessary changes on the cards as noted in the listing.

In addition to the controlling portion of this torsional response program, named TORRP, there are other subroutines. One, named PLNST, treats the planetary stage. Another, named MATIN, performs the matrix inversion that solves the simultaneous equations of the PLNST subroutine. Another, named BLOOP, applies to the branch treatment. The others, named CDIV, CDIV2, PANGF, AMPF, CAD, CSUB and CMPY, perform arithmetic operations.

COMPLEX STRUCTURAL DYNAMIC ANALYSIS COMPUTER PROGRAM USING STIFFNESS METHODS (D-82/C-51)

PROGRAM DESCRIPTION

Program D-82 calculates the natural frequencies and mode shapes of a complex structure composed of axial members, skins (membrane triangular), and beams. The program builds a stiffness matrix on tape of maximum order 3000 utilizing up to 5000 structural elements and 999 node points and 3 different materials. Using matrix arithmetic and assuming lumped masses, the stiffness matrix is reduced to a matrix of order 215. The dynamic matrix is formed using the masses, and natural frequencies and mode shapes are found. The program is completely general and may be used for most types of elastic systems. Axial, torsional, and bending motions may also be coupled by skewing the beam orienting node.

The input is composed of node points and their coordinates, structural elements including axials (areas), skins (thicknesses) and beams (moment of inertia, torsional moment of inertia and shear area), and masses for all degrees of freedom. The output includes the printout of the input, the initial gross stiffness matrix, the reduced stiffness matrix, the flexibility and dynamic matrices, and the eigenvalues and eigenvectors (natural frequencies and mode shapes). The mode shapes, as an option, may be used to calculate the damped forced response using program C-51. Vibratory loads are also calculated.

The C-51 program is a normal mode solution method which calculates vibration levels from external harmonic loads. In the solution, the natural modes from the D-82 analysis are used as generalized coordinates for representing the dynamic system. By simulating structural damping, the resulting second order nonhomogenous differential equations are solved. From these equations the program computes the response for each mode for sine and cosine forcing functions and obtains the total responses by a summation of the modal contributions. The maximum number of degrees of freedom is 215 and the maximum number of mode shapes is 30. The input of C-51 is in a standard loader format so that various cases may be run simultaneously and consists of a D-82 mode shape tape, exciting frequency, the eigenvectors desired, modal damping, and phased oscillatory forces and moments. The output of C-51 consists of a listing of the input plus the exciting frequency, forcing frequency ratios, amplification factors, modal phase angles, modal amplitudes, sine/cosine components, the resultants, phase angles and the "G" loading. In addition, vibratory forces and moments are calculated.

PROCEDURE FOR PREDICTING DAMPED FORCED RESPONSE

Helicopter vibration traceable to rotor excitation is of primary concern in the analysis of the airframe. Much effort has been expended on the development of computer techniques which assist in the design of structures with acceptable vibration levels. These procedures, initiated for the structure, are also applicable to the transmission system. The Unified Structural Analysis (or Damped Force Response) Computer Program (D-82) developed by J. Sciarra for the dynamic analysis of a helicopter fuselage has been extended for this purpose (References 17 and 18).

This program is capable of calculating the dynamic characteristics for a large complex structure. A typical helicopter transmission analytical model contains many structural elements, and the important elements of any analysis are:

1. Generation of a finite element structural idealization and discrete mass model.
2. Formulation of the internal components system stiffness matrix.
3. Reduction in the stiffness matrix of the unloaded nodal degrees of freedom to the loaded nodal degrees of freedom (mass points).
4. Dynamic matrix generalization combining mass and stiffness properties.
5. Determination of eigen solution.
6. Calculation of dynamic tooth loads from system torsional response analysis.
7. Formulation of dynamic equations and determination of a nodal representation of the transmission.
8. Solution for the damped response to be associated with the vibratory tooth loads.

---

17. Sciarra, J.J., A COMPUTER METHOD FOR DYNAMIC STRUCTURAL ANALYSIS USING STIFFNESS MATRIXES, Journal of Aircraft, Vol. 6, No. 1, January-February 1969.

18. Sciarra, J.J., USE OF THE FINITE ELEMENT DAMPED FORCED RESPONSE STRAIN ENERGY DISTRIBUTION FOR VIBRATION REDUCTION, Presented at the ARO-D Military Theme Review, The Helicopter and V/STOL Aircraft Research Conference, Moffett Field, California, September 1972.

The damped forced response of a multi-degree-of-freedom system is the normal mode solution to the following matrix equation:

$$\begin{bmatrix} M \end{bmatrix} \left\{ \ddot{x} \right\} + \begin{bmatrix} C \end{bmatrix} \left\{ \dot{x} \right\} + \begin{bmatrix} K \end{bmatrix} \left\{ x \right\} = \left\{ F_s \right\} \sin \Omega t + \left\{ F_c \right\} \cos \Omega t \quad (63)$$

where $\begin{bmatrix} M \end{bmatrix}$ = mass matrix

$\begin{bmatrix} C \end{bmatrix}$ = damping matrix

$\begin{bmatrix} K \end{bmatrix}$ = stiffness matrix

$\left\{ F_c \right\} \left\{ F_c \right\}$ = sine or cosine component of the exciting loads - lb, in.-lb

$\left\{ x \right\}$ = displacement-inches, rotation-radians

Ω = exciting frequency - rad/sec

t = time - sec

It is well known that the solution of the previous equation is

$$\left\{ x \right\} = \left\{ x_s \right\} \sin \Omega t + \left\{ x_c \right\} \cos \Omega t.$$

where $\left\{ x_s \right\}, \left\{ x_c \right\}$ = sine or cosine components of the displacement (or rotation) of the nodes of the structural element, inches, radians

Stiffness Method

Consider a complex structure, i.e., a transmission system, which for analysis will be idealized. Thus, junction or node points are selected where structural members meet, and such points will possess six degrees of freedom. Assume these latter members to be axial (resisting tension or compression), skin (resisting shear), and beams (opposing bending), such that each element contributes stiffness to the node at which it is attached. If forces or moments are applied to this structure, deflections or rotations will occur. The resulting set of simultaneous equations represent continuity and equilibrium of the collection of junctions and may be written as

$$F_i = \sum_{j=1}^{n} K_{ij} \; x_j \quad (i = 1,2,\ldots,n)$$

where n = number of degrees of freedom.

or, in matrix form,

$$\{F\} = [K]\{X\} \tag{64}$$

From physical consideration it may be shown that

$K_{ij} = K_{ji}$.

Dynamic Considerations

From equation (63), we may obtain the differential equations expressing the undamped free vibration of a system; thus,

$$[M]\{\ddot{X}\} + [K]\{X\} = 0 \tag{65}$$

Assuming a harmonic solution, it may be shown that equation (65) may be reduced to the form

$$\left[M^{-1}K-\lambda\right]\{X\} = 0$$

where λ = natural frequency squared (eigenvalue)

Utilization of the $M^{-1}K$ - matrix to ascertain the eigenvalues is cumbersome and numerically unpredictable; hence, an artifice known as reduction is employed to circumvent this difficulty. Partitioning equation (64), we may write

$$\left\{\begin{array}{c} F_1 \\ \hline F_2 \end{array}\right\} = \left[\begin{array}{c|c} K_{11} & K_{12} \\ \hline K_{21} & K_{22} \end{array}\right] \left\{\begin{array}{c} X_1 \\ \hline X_2 \end{array}\right\} \tag{66}$$

Then by lumping masses and moments of inertia at only "F_1" degrees of freedom, the inertia forces and torques at the "F_2" degrees of freedom are zero; equation (66) may be shown to yield

$$\{F_1\} = \left[K_{11} - K_{12}\, K_{22}^{\ -1}\, K_{21}\right]\{X_1\}$$

This technique, known as the reduction process, still admits deflections and rotations at the X_2's, and all stiffness contributions of the original K-matrix are included at the loaded nodes. However, the procedure yields a smaller dynamic matrix from which the system eigenvalues and eigenvectors are obtainable.

The collection of eigenvectors arranged in columns is called the modal matrix, $[\phi]$; hence, we let

$$\{x\} = [\phi]\{a\} \tag{67}$$

where ϕ = modal matrix (of order j by i)

a = modal displacements in row "i"

Then, by using equation (67) and premultiplying equation (65) by the transpose of the modal matrix, we have

$$\left[\phi^T \quad M\right]\left\{\phi \ddot{a}\right\} + \left[\phi^T \quad K\right]\left\{\phi a\right\} = 0$$

Assuming a harmonic solution, we find the matrix form of Rayleigh's quotient,

$$\lambda\left[\phi^T \quad M \quad \phi\right] = \left[\phi^T \ K \ \phi\right] \tag{68}$$

Now, if the eigenvalues are distinct, it may be shown that $\phi^T M \phi$ is a diagonal matrix. This diagonal matrix is called the effective mass matrix and by definition,

$$M_{eff} = \phi^T M \phi \quad \text{(a diagonal matrix)} \tag{69}$$

Substituting equation (69) into equation (68), we have

$$\phi^T K\phi = \lambda M_{eff} = K_{eff}$$

where K_{eff} = effective-stiffness matrix (a diagonal matrix)

Damped Forced Response due to Phased Excitation

Equation (63) represents the matrix equation for a complex structure. Premultiplying this equation by ϕ^T and introducing equation (67), we obtain

$$\left[\phi^T M \ \phi\right]\left\{\ddot{a}\right\} + \left[\phi^T \ C\phi\right]\left\{\dot{a}\right\} + \left[\phi^T \quad K\phi\right]\left\{a\right\}$$
$$= \left\{\phi^T \quad F_s\right\} \sin \Omega t + \left\{\phi^T \quad F_c\right\} \cos \Omega t \tag{71}$$

Assuming modal damping, we may write

$$\left[\phi^T \ C\phi\right] = \left[2 \ \zeta_i \quad M_{eff} \ \lambda^{1/2}\right] = \left[C_{effi}\right] \tag{72}$$

where ζ_i = critical modal damping coefficient

C_{effi} = effective-damping matrix (a diagonal matrix)

Further, let us define

$$\left\{F_{eff\ s}\right\} = \left\{\phi^T F_s\right\}$$
$$\left\{F_{eff\ c}\right\} = \left\{\phi^T F_c\right\} \qquad (73)$$

Substituting equations (69), (70), (72), and (73) into equation (71), we have

$$\left[M_{eff\ i}\right]\left\{\ddot{a}\right\} + \left[C_{eff_i}\right]\left\{\dot{a}_i\right\} + \left[K_{eff_i}\right]\left\{a_i\right\} =$$
$$\left\{F_{eff_{si}}\right\} \sin\Omega t + \left\{F_{eff_{ci}}\right\} \cos\Omega t \qquad (74)$$

If it is assumed that the undamped natural frequency equals the damped natural frequency, because of small structural damping, and since equations (67), (70), and (72) are diagonal, then equation (74) represents a set of uncoupled, second order, differential equations.

It may be shown that t' . solutions of equation (74) are given by

$$\left\{a_i\right\} = \left\{G_i\right\} \sin\Omega t - \left\{H_i\right\} \cos\Omega t \qquad (75)$$

where
$$\left\{G_i\right\} = \left\{R_{s_i} \cos \phi_i + R_{c_i} \sin \phi_i\right\}$$
$$\left\{H_i\right\} = \left\{R_{s_i} \sin \phi_i - R_{c_i} \cos \phi_i\right\}$$
$$R_{s_i} = \mu_i \; F_{s_i} / K_{eff_i}$$
$$R_{c_i} = \mu_i \; F_{c_i} / K_{eff_i}$$
$$\phi_i = \mathrm{TAN}^{-1}\left[2 \zeta_i \beta_i / (1-\beta_i^2)\right]$$
$$\mu_i = 1/\left[(1-\beta_i^2)^2 + (2 \zeta_i \beta_i)^2\right]^{1/2}$$
$$\beta_i = \Omega / \sqrt{\lambda}$$

Therefore, by equations (67) and (75) the rotations and displacements are expressible as

$$\left\{X_i\right\} = \left\{P_j\right\} \sin\Omega t - \left\{S_j\right\} \cos\Omega t \qquad (76)$$

where
$$P_j = \phi \; G_i$$
$$S_j = \phi \; H_i$$

Employing trigonometric relations, equation (76) may be written in the form,

$$\left\{X_i\right\} = \left\{(P^2_j + S_j^2)^{1/2} \sin(\Omega t - \psi_j)\right\}$$

$$\text{where } \psi_j = \text{TAN}^{-1} (S_j / P_j)$$

To complete the analysis of the internal components, the total displacements of the structure are found corresponding to the dynamic tooth forces. These loads are determined by the torsional response computer program TORRP (R-32), in which the exciting frequencies are taken to be multiples of the mesh frequencies.

COMPUTER MODELING OF DYNAMIC COMPONENTS

Modeling of the dynamic components of the system is an involved, lengthy procedure. It is important to remember that growth and improvements should be provided for in the development of the model. The procedure for modeling shafts in the Unified Structure Analysis Program (D-82), which is a finite element analysis, is outlined below:

1. Shafts are divided into small cylindrical sections, and the lengths of these sections are determined by the variation of the shaft diameter. Conical sections are also divided into small cylinders which vary in a discontinuous manner.

2. Physical properties (mass, inertia, cross-sectional area, etc.) are calculated for each individual cylinder.

3. Masses and polar moments of inertia are averaged between adjacent stations, if computer capacity allows; otherwise, an equivalent mass and inertia are fixed to selected positions along the shaft length.

4. Gear meshes are represented by four masses located at the appropriate pitch diameter. It is important, however, that the mass concentrated at the gear nodes be representative of the actual mass distribution. These masses are connected to the main shaft by beams exhibiting elastic properties similar to the actual gears. Additional beams are provided to insure that the four masses remain at equal distance from one another.

5. Radial stiffness of bearings is included in the analysis by pairs of linear and torsional springs. To allow for nonuniform stiffness characteristics, two mutually perpendicular springs of different stiffness are required at

each bearing location. Thrust stiffness is represented by a linear spring acting in the axial direction. These stiffnesses are a function of torque and were calculated on the basis of a scheme developed by A. B. Jones (Reference 19).

6. Contacting gears are represented by a spring exhibiting an elastic behavior similar to the tooth stiffness as calculated by Computer Program GGEAR (R-67).

7. A synchronizing shaft is modeled as a torsional spring determined by the geometry of the device.

8. Existence of planetary stages is represented by a torsional spring whose stiffness is extracted from the Computer Program TORRP (R-32).

9. The four planets of the lower stage planetary system are depicted by four equally spaced linear springs. Stiffness of these springs was determined to be approximately one-half of the calculated carrier post stiffness.

This model, developed for the CH-47 forward transmission, is illustrated in Figure 26. The necessity of coupling the shafts was dictated by comparative analysis of the experimental and initial analytical results.

---

19. Jones, A.B., A GENERAL THEORY FOR ELASTICALLY CONSTRAINED BALL AND RADIAL ROLLER BEARINGS UNDER ARBITRARY LOAD AND SPEED CONDITIONS, ASME Paper 59-LUB-10.

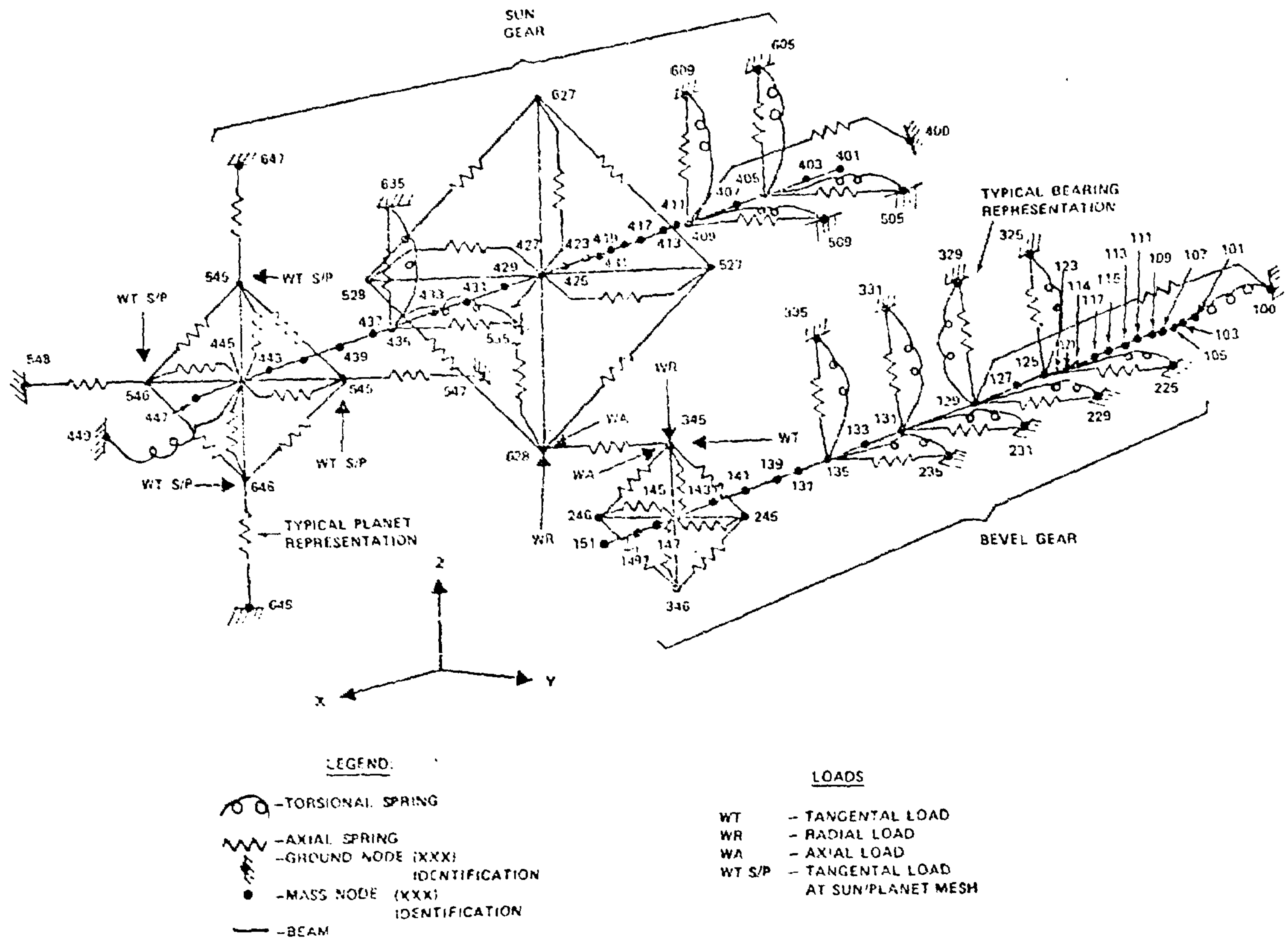


Figure 26. Mathematical Model of CH-47 Transmission.

PROGRAM USAGE - INPUT PREPARATION

There are four basic dynamic analysis input sheets:

- Control Numbers - Figure 27
- Coordinates and Nodal Boundary Conditions (Action Table) - Figure 28
- Structural Elements (Sat Table) - Figure 29
- Masses and Mass Moments of Inertia - Figure 30

Figures 31 and 32 are used for the damped forced response. Assuming that the idealization of the structure is complete, fill in the control cards (Figure 33) as follows:

- Input the title in the header block.
- Count the number of nodes and enter with decimal in the block "# nodes". The maximum number of nodes and the maximum number allowable for a node number is 999. This count must be correct.
- Count the number of different structural elements and enter with decimal in the block "# SATS". The maximum number of structural elements is 5000. This count must be correct.
- Count the number of retained degrees of freedom for the node points. This corresponds to the number of "2's" in the action table input sheets. The maximum number of retained degrees of freedom is 215. This count must be right. Each retained degree of freedom must have a mass or mass moment of inertia associated with it.
- Count the number of reduced degrees of freedom and enter in "# REDUCED" box. This corresponds to the number of "1's" in the action table input sheets. The sum of the reduced and retained degrees of freedom must be less than 3000. If a nodal degree of freedom is reduced, the node will be allowed motion, but there will be no mass or moment of inertia for that particular degree of freedom. The count of the number of reduced degrees of freedom must be correct or the program will fail.

Engineer __________________ INPUT SHEET Page _____

Date ______________ CONTROL NUMBERS

SCIARRA/WOLFINGER USA METHOD

TITLE CARD PLEASE PUNCH

7	72

0001	# NODES (MAX. = 999.)		
	# SATS (MAX. = 3000.)		
	# RETAIN D.O.F.'s (MAX. = 215.)		
	# REDUCED D.O.F.		
	# MATERIAL CODES (MAX.=3.)		
0006	MAX. NODE # (MAX. = 999.)		
	YOUNG'S MODULUS 1ST MATERIAL		
	YOUNG'S MODULUS 2ND MATERIAL		
	YOUNG'S MODULUS 3RD MATERIAL		
	POISSON'S RATIO 1ST MATERIAL		
0011	POISSON'S RATIO 2ND MATERIAL		
	POISSON'S RATIO 3RD MATERIAL		
	# VARIATIONS (MASS)	1.	
	# EIGENVECTORS		
	# MASS GROUPS		
0016	SCALE FACTOR	1.	
	SAVE TAPE (1.=YES 0.=NO)	1.	NOTE: MAKES 2 TAPES
0018	PRINT STRAIN ENERGY (1.=YES)	0.	
	#SPECIAL ELEMENTS	0.	
	1.=DON'T PRINT INPUT	0.	
0021	1.=DON'T PRINT FULL K	0.	
	1.=DON'T PRINT REDUCED K	0.	
	CALCULATE EIGENVECTORS > THIS.		
0026	1. = FLEXIBILITY MATRIX = K^{-1}		
	0=D82, 1.=C51 Only		
0031	0=D82, 1.=D82 and C51		
	ROW NORMALIZED TO UNITY		

Figure 27. Input Sheet - Control Numbers.

AT TABLE (COORDINATES AND BOUNDARY CONDITIONS) (SECOND ITEM)

1–3	4–13	14–23	24–33	34–36	37–39	40–63
NODE	X	Y	Z	DOF	NOS	

PUT A "9" PUNCH IN COL 72 OF THE LAST CARD OF THE "ATS".

Figure 28. Coordinates and Nodal Boundary Conditions.

SAT TABLE (STRUCTURAL CONNECTORS)

(THIRD ITEM)

TYPE	NØDES			M/C	STRINGER AREA A_i or SHEAR THICKNESS OF SKIN t_s or MOMENT OF INERTIA OF BEAM I_y	THICKNESS OF SKIN t_x or TORSIONAL MOMENT OF INERTIA OF BEAM J_x	THICKNESS OF SKIN t_y or SHEAR WEB AREA OF BEAM A_w	

TYPE: 002 = AXIAL, 003 = SKIN, 004 = BEAM; PUT "9" PUNCH, COL 72, LAST SAT

Figure 29. Structural Elements.

MASSES OR MOMENTS OF INERTIA (4TH ITEM)

1 4	6 15	16 25	26 35	36 45	46 55	56 65	66 77

NØDE
LEFT ADJUST
NUMBERS

1. "9" PUNCH, COL 80, LAST MASS CARD
2. "99999", COLS 1-5, AFTER LAST MASS

Figure 30. Masses and Mass Moments of Inertia.

PAGE_____

DAMPED FORCED RESPONSE

NAME ____________

DATE ____________

0001		Ω RAD/SEC
0002	1.	
0003		
0004		
0005		EIGENVECTORS DESIRED
0006		
0007		
0008		
0009		
0010		
0011		
0012		
0013		
0014		
0015		
0016		
0017		
0018		
0019		
0020		
0021		
0022		
0023		
0024		
0025		
0026		
0027		
0028		
0029		
0030		
0031		
0032		
0033		
0034		
0035		CRITICAL DAMPING
0036		
0037		
0038		
0039		
0040		
0041		
0042		
0043		
0044		
0045		
0046		
0047		

0048	
0049	
0050	
0051	
0052	
0053	
0054	
0055	
0056	
0057	
0058	
0059	
0060	
0061	
0062	
0063	
0064	
0065	
0066	1.

88888 (LAST CARD)

Figure 31. Damped Forced Response Parameters.

NAME ______________ PAGE ____

PHASED EXTERNAL EXCITING LOADS = F_s SINΩt + F_c COSΩt

F_s

88888

SINE COMPONENT (F_s), (lbs or inch/lbs) maximum of 215.

F_c

88888

COSINE COMPONENT (F_c), maximum of 215.

Figure 32. Exciting Loads.

Engineer______________ INPUT SHEET Page____

Date____________ CONTROL NUMBERS

SCIARRA/WOLFINGER USA METHOD

TITLE CARD PLEASE PUNCH

7	SCIARRA TEST CASE	72

0001	# NODES (MAX. = 999.)	18.
	# SATS (MAX. = 3000.)	35.
	# RETAIN D.O.F.'s (MAX. = 215.)	30.
	# REDUCED D.O.F.	20.
	# MATERIAL CODES (MAX.=3.)	1.
0006	MAX. NODE # (MAX. = 999.)	600.
	YOUNG'S MODULUS 1ST MATERIAL	30.E6
	YOUNG'S MODULUS 2ND MATERIAL	0.
	YOUNG'S MODULUS 3RD MATERIAL	0.
	POISSON'S RATIO 1ST MATERIAL	.33
0011	POISSON'S RATIO 2ND MATERIAL	0.
	POISSON'S RATIO 3RD MATERIAL	0.
	# VARIATIONS (MASS)	1.
	# EIGENVECTORS	5.
	# MASS GROUPS	10.
0016	SCALE FACTOR	1.
	SAVE TAPE (1.=YES 0.=NO)	1.
0018	PRINT STRAIN ENERGY (1.=YES)	0.
	#SPECIAL ELEMENTS	0.
	1.=DON'T PRINT INPUT	0.
0021	1.=DON'T PRINT FULL K	0.
	1.=DON'T PRINT REDUCED K	0.
	CALCULATE EIGENVECTORS > THIS.	0.
0026	1. = FLEXIBILITY MATRIX = K^{-1}	0.
	0=D82, 1.=C51 Only	0.
0031	0=D82, 1.=D82 and C51	1.
	ROW NORMALIZED TO UNITY	650.

NOTE: MAKES 2 TAPES

Figure 33. Input Sheet - Control Numbers.

In locations 5 and 6, numbers indicated on the input sheet are used. In location 7, enter Young's modulus. "E" notation may be used. In location 10, enter Poisson's ratio. In location 14, enter the number of eigenvectors desired (maximum of 30). In location 15, enter the number of mass points. Location 25 is the lower bound for the eigenvalues for which eigenvectors are desired. Location 26 is the indicator for forming the inverse of the stiffness matrix. Locations 13, 16, 17, 18, and 31 are to be input as indicated on the input sheet. All other input quantities are ignored for the present.

The second input sheet (Figure 34) is the "AT" or action table. In columns 1 to 3 enter the node number (maximum number is 999). If the node number is "5", enter as 005. In columns 4 to 33, enter the x, y, and z coordinates of the node point. Use a decimal point. In columns 34 to 39, enter the boundary conditions for the node point as follows:

- "0" means delete the degree of freedom. This means no motion is allowed for that particular degree of freedom. It also means no rotation in the case of a node point connecting only by axial or skin members. It is also used for degrees of freedom associated with beams oriented in some specialized direction. It is also input for nodes used only for orienting beams, but with no structural duty.

- "1" means reduce the degree of freedom. This means that <u>no</u> mass or moment of inertia is to be associated with that particular degree of freedom. However, motion will be allowed and the element stiffness will be merged into the final retained matrix.

- "2" means retain the degree of freedom. This implies that a mass or mass moment of inertia will be associated with that particular degree of freedom.

The order that these boundary conditions are input is x, y, z, Θ_x, Θ_y, and Θ_y. The last three represent rotations about the respective axes.

The third table (Figure 35) is the "SAT" (Structural Assemblage Table) and is filled out as follows:

- In Column 3, put "2" if the element is an axial member, "3" if a skin (triangular) element, and "4" if a beam element.

- In columns 4 to 12, insert the node numbers associated with the member. If it is an axial (stringer) element,

AT TABLE (COORDINATES AND BOUNDARY CONDITIONS) (SECOND ITEM)

1–3	X (4–13)	Y (14–23)	Z (24–33)	DØF (34–36)	NØS (37–39)	40–63
001	0.0	0.0	10.25	000	000	
002	0.0	0.0	0.0	000	000	
003	20.	0.0	10.25	222	101	
004	20.	0.0	0.0	222	101	
005	40.	0.0	10.25	222	101	
006	40.	0.0	0.0	222	101	
007	60.	0.0	10.25	222	101	
008	60.	0.0	0.0	222	101	
009	80.	0.0	10.25	222	101	
010	80.	0.0	0.0	222	101	
011	100.	0.0	10.25	222	101	
012	100.	0.0	0.0	222	101	
013	100.	1.0	10.30	000	000	
014	100.	1.0	.05	000	000	
015	20.	1.0	10.25	000	000	
016	40.	1.0	10.25	000	000	
017	60.	1.0	10.25	000	000	
018	80.	1.0	10.25	000	000	

"9" PUNCH COL. 72

PUT A "9" PUNCH IN COL 72 OF THE LAST CARD OF THE "ATS".

Figure 34. Coordinates and Nodal Boundary Conditions.

SAT TABLE (STRUCTURAL CONNECTORS) (THIRD ITEM)

TYPE	NODES			M/C	STRINGER AREA / SHEAR THICKNESS OF SKIN / MOMENT OF INERTIA OF BEAM: A_s or t_s or I_y	THICKNESS OF SKIN (X) / TORSIONAL MOMENT OF INERTIA OF BEAM: t_x or J_x	THICKNESS OF SKIN (Y) / SHEAR WEB AREA OF BEAM: t_y or A_w	
002	001	003			6.99			
	003	005			6.77			
	005	007			6.88			
	007	009			6.88			
	009	011			6.88			
	002	004			6.88			
	004	006			6.88			
	006	008			6.88			
	008	010			6.88			
002	010	012			6.88			
003	001	002	003		.415	.415	.415	
	002	003	004		.415	.415	.415	
	003	004	005		.415	.415	.415	
	004	005	006		.415	.415	.415	
	005	006	007		.415	.415	.415	
	006	007	008		.415	.415	.415	
	007	008	009		.415	.415	.415	
	008	009	010		.415	.415	.415	
	009	010	011		.415	.415	.415	
003	010	011	012		.415	.415	.415	
004	001	003	013		58.25	58.47	6.88	
	003	005	013		58.25	58.47	6.88	
	005	007	013		58.25	58.27	6.88	
	007	009	013		58.25	58.27	6.88	
	009	011	013		58.25	58.27	6.88	
004	002	004	014		58.25	58.27	6.88	
004	004	006	014		58.25	58.27	6.88	
	006	008	014		58.25	58.27	6.88	
	008	010	014		58.25	58.27	6.88	
	010	012	014		58.25	58.27	6.88	
	003	004	015		.12	276.0	8.30	
	005	006	016		.12	276.0	8.30	
	007	008	017		.12	276.0	8.30	
	009	010	018		.12	276.0	8.30	
004	011	012	013		.12	276.0	8.30	"9" PUNCH COL 72 →

TYPE: 002 = AXIAL, 003 = SKIN, 004 = BEAM; PUT "9" PUNCH, COL 72, LAST SAT

Figure 35. Structural Elements.

these would be the two end nodes only. Leave the last node box empty. If the member is a triangular (shear panel) element, input the three node points. If the member is a beam, input the end node points of the beam. In the third box for the beam enter an orienting node. The two end node points define the $\bar{x}$ - axis (neutral axis) of the beam. The orienting node point defines the $\bar{z}$ - axis of the beam. The $\bar{y}$ - axis would be perpendicular to the $\bar{x}$ - $\bar{z}$ plane.

- For an axial member, insert the cross-sectional area in columns 14 to 23. Leave columns 24 to 43 empty. For the skin member, input the thickness resisting in-plane shear t_s, the thickness in the $\bar{x}$-direction (from node 1 to node 2) $t_{\bar{x}}$, and the thickness in the $\bar{y}$ direction (perpendicular to the $\bar{x}$ direction) $t_{\bar{y}}$. For a beam, input the moment of inertia about the $\bar{y}$ - axis, $I_{\bar{y}}$, the torsional moment of inertia about the $\bar{x}$ - axis, $J_{\bar{x}}$, and the cross-sectional area effective in resisting shear, A_w.

The fourth input sheet (Figure 36) is for the masses or mass moments of inertia that are associated with a retained degree of freedom. In columns 1 to 4, input the node number. For example input 0125, 0001. Note that 4 digits are used. Column 5 is skipped. Columns 6 to 65 have room for 6 numbers or 6 degrees of freedom for any node. Use decimals. If less than 6 degrees of freedom are associated with a node point, only input masses or moments of inertia for the exact number of degrees of freedom. Move all numbers to the left. For example: 0023 1. 10. might be a mass in the x - direction and a moment of inertia about the z - axis for node 23. The numbers 1. and 10. would appear in blocks 1 and 2.

In order to calculate the damped forced response, sheets 5 (Figure 37) and 6 (Figure 38) are used.

On Figure 37, location 0001 inputs the exciting frequency. Repeat this number in location 0003. In locations 5 to 34, enter the number of the eigenvector desired (e.g., 1., 2., 3., 4. for the first to fourth modes). In locations 35 to 64, input the assumed modal damping for each mode (e.g., 0.03).

On Figure 38, input the external sine and cosine loads. In the first column, input the row number of the degree of freedom being excited (e.g., row 0017 for exciting node 10 in the x - direction). This corresponds to the location in the stiffness matrix of the degree of freedom involved.

MASSES OR MOMENTS OF INERTIA (4TH ITEM)

1 4	6 15	16 25	26 35	36 45	46 55	56 65	66 77
0003	.13	.13	.13				
0004	.13	.13	.13				
0005	.13	.13	.13				
0006	.13	.13	.13				
0007	.13	.13	.13				
0008	.13	.13	.13				
0009	.13	.13	.13				
0010	.13	.13	.13				
0011	.13	.13	.13				
0012	.13	.13	.13				
99999						"9" PUNCH COL 80	

NØDE
LEFT ADJUST
NUMBERS

1. "9" PUNCH, COL 80, LAST MASS CARD
2. "99999", COLS 1-5, AFTER LAST MASS

Figure 36. Masses and Mass Moments of Inertia.

PAGE______

DAMPED FORCED RESPONSE

NAME ____________

DATE ____________

0001	62.8318	Ω RAD/SEC
0002	1.	
0003	62.8318	
0004		
0005	1.	EIGENVECTORS DESIRED
0006	2.	
0007	3.	
0008	4.	
0009	5.	
0010		
0011		
0012		
0013		
0014		
0015		
0016		
0017		
0018		
0019		
0020		
0021		
0022		
0023		
0024		
0025		
0026		
0027		
0028		
0029		
0030		
0031		
0032		
0033		
0034		
0035	.02	CRITICAL DAMPING
0036	.02	
0037	.02	
0038	.02	
0039	.02	
0040		
0041		
0042		
0043		
0044		
0045		
0046		
0047		

0048	
0049	
0050	
0051	
0052	
0053	
0054	
0055	
0056	
0057	
0058	
0059	
0060	
0061	
0062	
0063	
0064	
0065	
0066	1.
88888	(LAST CARD)

Figure 37. Damped Forced Response Parameters.

NAME ____________ PAGE ____

PHASED EXTERNAL EXCITING LOADS = $F_s \sin\Omega t + F_c \cos\Omega t$

F_s

0027	1000.

88888

Node 11 Z

SINE
COMPONENT (F_s),
(lbs or inch/lbs)
maximum of 215.

F_c

0027	1000.

88888

COSINE
COMPONENT (F_c),
maximum of 215.

NOTES:
1. For static deflection use F_c only and set F_s's and Ω to zero.
2. After this run, these loads are erased and must be input again for subsequent runs.

Figure 38. Exciting Loads.

PROGRAM OUTPUT

The output data generated for the test problem from program D-82/C-51 is shown in Figure 39, along with the input data previously described. In addition to the above data, program D-82/C-51 will produce a tape of mode shapes containing modal information for the transmission internal components which is used as input into program S68. A listing of the actual internal components shafting model is shown in Figure 40.

```
SCIARRA / JACKSON                                                         PAGE   1
                         FUSELAGE ANALYSIS PROGRAM-HELAN FUD
                           DF200  SCIARRA / JACKSON / JOANNE

          NO. OF NODES =      18
          NO. OF STRUCTURAL ELEMENTS =     35
          NO. OF RETAINED DOFS =    30
          NO. OF REDUCED DOFS =      20
          NO. OF MATERIAL PROPERTIES =      1
          MAX. NODE NO. USED =     600
          NO. OF DELETED DOFS =      58
          FULL SIZE OF MATRIX BEFORE REDUCTION =     50
          NO. OF VARIATIONS=      1
          NO. OF EIGENVECTORS=      5
          NO. OF MASS GROUPS=     10
          NO. OF SPECIAL ELEMENTS=     0
          ****************************************************************
          *         THE TAPE REGISTER HAS PRESERVED IS ----               *
          ****************************************************************

                    MATERIAL PROPERTIES
          MODULUS                              POISSON RATIO
          [illegible]                             0.3300

              INPUT OPTIONS
          INPUT                YES
          FULL [illegible]     YES
          [illegible]          YES

          EIGENVECTORS CALCULATED FOR EIGENVALUES GREATER THAN 0.0
                                   ACTION TABLE
```

NODE	X	Y	Z	BONE	BTAB
1	0.0	0.0	10.2500	0	0
2	0.0	0.0	0.0	0	0
3	20.0000	0.0	10.2500	222	101
4	20.0000	0.0	0.0	222	101
5	40.0000	0.0	10.2500	222	101
6	40.0000	0.0	0.0	222	101
7	60.0000	0.0	10.2500	222	101
8	60.0000	0.0	0.0	222	101
9	80.0000	0.0	10.2500	222	101
10	80.0000	0.0	0.0	222	101
11	100.0000	0.0	10.2500	222	101
12	100.0000	0.0	0.0	222	101
13	100.0000	1.0000	10.5000	0	0
14	100.0000	1.0000	0.0500	0	0
15	20.0000	1.0000	10.2500	[illegible]	0
16	40.0000	1.0000	10.2500	0	0
17	60.0000	1.0000	10.2500	0	0
18	80.0000	1.0000	10.2500	0	0

```
                    MAP-RETAINED

     3   3   3   4   4   4   5   5   5   6   6   6   7   7   7   8
     8   8   9   9   9  10  10  10  11  11  11  12  12  12

                    MAP-REDUCED

     3   3   4   4   5   5   6   6   7   7   8   8   9   9  10  10
    11  11  12  12
```

Figure 39. Sample D-82/C-51 Output.

```
SCIARRA TEST CASE
    SCIARRA - JACKSON - JUSTINE USA MATRIX METHOD OF STRUCTURAL ANALYSIS
-------------------------------------------------------------------------------------
                              AXIAL ELEMENT
 N1=   1  N2=   3  MATERIAL CODE= 1  AREA= 0.68800D 01   0.0          0.0
-------------------------------------------------------------------------------------
                              AXIAL ELEMENT
 N1=   3  N2=   5  MATERIAL CODE= 1  AREA= 0.68800D 01   0.0          0.0
-------------------------------------------------------------------------------------
                              AXIAL ELEMENT
 N1=   5  N2=   7  MATERIAL CODE= 1  AREA= 0.68800D 01   0.0          0.0
-------------------------------------------------------------------------------------
                              AXIAL ELEMENT
 N1=   7  N2=   9  MATERIAL CODE= 1  AREA= 0.68800D 01   0.0          0.0
-------------------------------------------------------------------------------------
                              AXIAL ELEMENT
 N1=   9  N2=  11  MATERIAL CODE= 1  AREA= 0.68800D 01   0.0          0.0
-------------------------------------------------------------------------------------
                              AXIAL ELEMENT
 N1=   2  N2=   4  MATERIAL CODE= 1  AREA= 0.68800D 01   0.0          0.0
-------------------------------------------------------------------------------------
                              AXIAL ELEMENT
 N1=   4  N2=   6  MATERIAL CODE= 1  AREA= 0.68800D 01   0.0          0.0
-------------------------------------------------------------------------------------
                              AXIAL ELEMENT
 N1=   6  N2=   8  MATERIAL CODE= 1  AREA= 0.68800D 01   0.0          0.0
-------------------------------------------------------------------------------------
                              AXIAL ELEMENT
 N1=   8  N2=  10  MATERIAL CODE= 1  AREA= 0.68800D 01   0.0          0.0
-------------------------------------------------------------------------------------
                              AXIAL ELEMENT
 N1=  10  N2=  12  MATERIAL CODE= 1  AREA= 0.68800D 01   0.0          0.0
-------------------------------------------------------------------------------------
                                TRIANGULAR SKIN ELEMENT
 N1=   1  N2=   2  N3=   3  MATERIAL CODE= 1  TS= 0.4150  TX= 0.4150
 TY= 0.4150   0.0          0.0          0.0          0.0
-------------------------------------------------------------------------------------
                                TRIANGULAR SKIN ELEMENT
 N1=   2  N2=   3  N3=   4  MATERIAL CODE= 1  TS= 0.4150  TX= 0.4150
 TY= 0.4150   0.0          0.0          0.0          0.0
-------------------------------------------------------------------------------------
                                TRIANGULAR SKIN ELEMENT
 N1=   3  N2=   4  N3=   5  MATERIAL CODE= 1  TS= 0.4150  TX= 0.4150
 TY= 0.4150   0.0          0.0          0.0          0.0
-------------------------------------------------------------------------------------
                                TRIANGULAR SKIN ELEMENT
 N1=   4  N2=   5  N3=   6  MATERIAL CODE= 1  TS= 0.4150  TX= 0.4150
 TY= 0.4150   0.0          0.0          0.0          0.0
-------------------------------------------------------------------------------------
                                TRIANGULAR SKIN ELEMENT
 N1=   5  N2=   6  N3=   7  MATERIAL CODE= 1  TS= 0.4150  TX= 0.4150
 TY= 0.4150   0.0          0.0          0.0          0.0
-------------------------------------------------------------------------------------
                                TRIANGULAR SKIN ELEMENT
 N1=   6  N2=   7  N3=   8  MATERIAL CODE= 1  TS= 0.4150  TX= 0.4150
 TY= 0.4150   0.0          0.0          0.0          0.0
-------------------------------------------------------------------------------------
                                TRIANGULAR SKIN ELEMENT
 N1=   7  N2=   8  N3=   9  MATERIAL CODE= 1  TS= 0.4150  TX= 0.4150
 TY= 0.4150   0.0          0.0          0.0          0.0
-------------------------------------------------------------------------------------
                                TRIANGULAR SKIN ELEMENT
 N1=   8  N2=   9  N3=  10  MATERIAL CODE= 1  TS= 0.4150  TX= 0.4150
 TY= 0.4150   0.0          0.0          0.0          0.0
-------------------------------------------------------------------------------------
                                TRIANGULAR SKIN ELEMENT
 N1=   9  N2=  10  N3=  11  MATERIAL CODE= 1  TS= 0.4150  TX= 0.4150
 TY= 0.4150   0.0          0.0          0.0          0.0
-------------------------------------------------------------------------------------
```

SCIARRA - JACKSON - JOANNE USA METHOD OF STRUCTURAL ANALYSIS

TRIANGULAR SKIN ELEMENT

N1= 13 N2= 11 N3= 12 MATERIAL CODE= 1 TS= 0,4150 TX= 0,4150

TY= 0,4150 0,0 0,0 0,0 0,0

BEAM ELEMENT

N1= 1 N2= 3 N3= 13 MATERIAL CODE= 1 IY= 0,58250D 02 JX= 0,58470D 02

A= 0,668000 01 0,0 0,0

BEAM ELEMENT

N1= 3 N2= 5 N3= 13 MATERIAL CODE= 1 IY= 0,58250D 02 JX= 0,58470D 02

A= 0,[illegible] 01 0,0 0,0

BEAM ELEMENT

N1= 5 N2= 7 N3= 13 MATERIAL CODE= 1 IY= 0,58250D 02 JX= 0,58270D 02

A= 0,[illegible] 01 0,0 0,0

BEAM ELEMENT

N1= 7 N2= 9 N3= 13 MATERIAL CODE= 1 IY= 0,58250D 02 JX= 0,58270D 02

A= 0,[illegible] 01 0,0 0,0

BEAM ELEMENT

N1= 9 N2= 11 N3= 13 MATERIAL CODE= 1 IY= 0,58250D 02 JX= 0,58270D 02

A= 0,[illegible] 01 0,0 0,0

BEAM ELEMENT

N1= 2 N2= 4 N3= 14 MATERIAL CODE= 1 IY= 0,58250D 02 JX= 0,58270D 02

A= 0,[illegible] 01 0,0 0,0

BEAM ELEMENT

N1= 4 N2= 6 N3= 14 MATERIAL CODE= 1 IY= 0,58250D 02 JX= 0,58270D 02

A= 0,[illegible] 01 0,0 0,0

BEAM ELEMENT

N1= [illegible] N2= 8 N3= 14 MATERIAL CODE= 1 IY= 0,58250D 02 JX= 0,58270D 02

A= 0,[illegible] 01 0,0 0,0

BEAM ELEMENT

N1= 8 N2= [illegible] N3= 14 MATERIAL CODE= 1 IY= 0,58250D 02 JX= 0,58270D 02

A= 0,[illegible] 01 0,0 0,0

BEAM ELEMENT

N1= 10 N2= 12 N3= 14 MATERIAL CODE= 1 IY= 0,58250D 02 JX= 0,58270D 02

A= 0,588000 01 0,0 0,0

BEAM ELEMENT

N1= 3 N2= 4 N3= 15 MATERIAL CODE= 1 IY= 0,12000D 00 JX= 0,27600D 03

A= 0,[illegible] 01 0,0 0,0

BEAM ELEMENT

N1= [illegible] N2= 6 N3= 16 MATERIAL CODE= 1 IY= 0,12000D 00 [illegible] 0,27600D 03

A= 0,[illegible] 0,0 0,0

BEAM ELEMENT

N1= 7 N2= 8 N3= 17 MATERIAL CODE= 1 IY= 0,12000D 00 JX= 0,27600D 03

A= 0,630000 01 0,0 0,0

BEAM ELEMENT

N1= 9 N2= 10 N3= 18 MATERIAL CODE= 1 IY= 0,12000D 00 JX= 0,27600D 03

A= 0,830000 01 0,0 0,0

BEAM ELEMENT

N1= 11 N2= 12 N3= 13 MATERIAL CODE= 1 IY= 0,12000D 00 JX= 0,27600D 03

A= 0,[illegible] 01 0,0 0,0

TOTAL NUMBER OF STRUCTURAL ELEMENTS 35

NO. OF AXIAL ELEMENTS 10

NO. OF SKI ELEMENTS 10

NO. OF BEAM ELEMENTS 15

SCIARRA - JACKSON - JOANNE USA METHOD OF STRUCTURAL ANALYSIS

RETAINED STIFFNESS MATRIX

MATRIX 1 30 ROWS 30 COLUMNS STORAGE MODE 1 PAGE 1

COLUMN	1	2	3	4
ROW 1	0.369330D 08	0.0	-0.464552D 07	-0.913259D 07
ROW 2	0.0	0.278567D 07	0.137389D 06	0.0
ROW 3	-0.464552D 07	0.137389D 06	0.296671D 08	0.464552D 07
ROW 4	-0.913259D 07	0.0	0.464552D 07	0.369330D 08
ROW 5	0.0	-0.186563D 06	-0.743391D 04	0.0
ROW 6	0.464552D 07	-0.743391D 04	-0.272016D 08	-0.464552D 07
ROW 7	-0.139002D 08	0.0	0.230530D 07	0.0
ROW 8	0.0	-0.156131D 07	-0.779609D 05	0.0
ROW 9	0.254023D 07	-0.779609D 05	-0.120326D 07	-0.464552D 07
ROW 10	0.0	0.0	0.0	-0.139002D 08
ROW 11	0.0	0.245664D 04	0.380112D 02	0.0
ROW 12	0.0	0.380112D 02	0.190056D 01	0.230530D 07
ROW 13	0.0	0.0	0.0	0.0
ROW 14	0.0	0.365620D 06	0.183532D 05	0.0
ROW 15	0.0	0.183532D 05	0.917662D 03	0.0
ROW 16	0.0	0.0	0.0	0.0
ROW 17	0.0	0.145005D 06	0.717799D 04	0.0
ROW 18	0.0	0.717799D 04	0.358900D 03	0.0
ROW 19	0.0	0.0	0.0	0.0
ROW 20	0.0	-0.445025D 05	-0.216084D 04	0.0
ROW 21	0.0	-0.216084D 04	-0.108042D 03	0.0
ROW 22	0.0	0.0	0.0	0.0
ROW 23	0.0	-0.278144D 05	-0.145500D 04	0.0
ROW 24	0.0	-0.145500D 04	-0.727501D 02	0.0
ROW 25	0.0	0.0	0.0	0.0
ROW 26	0.0	0.319486D 04	0.220167D 03	0.0
ROW 27	0.0	0.220167D 03	0.110084D 02	0.0
ROW 28	0.0	0.0	0.0	0.0
ROW 29	0.0	0.478432D 04	0.178892D 03	0.0
ROW 30	0.0	0.178492D 03	0.894461D 01	0.0

[illegible] - JACKSON - JUA[illegible] USA MATRIX METHOD OF STRUCTURAL ANALYSIS

[illegible] 1 30 ROWS 30 COLUMNS STORAGE MODE 1 PAGE 2

COLUMN ,		5	6	7	8
ROW	1	0.0	0.4645520 07	-0.1390020 08	0.0
ROW	2	-0.1865650 06	-0.7433910 04	0.0	-0.1561310 07
ROW	3	-0.7433910 04	-0.2726160 08	0.2305300 07	-0.7798090 05
ROW	4	0.0	-0.4645520 07	0.0	0.0
ROW	5	0.2785670 07	0.1373690 06	0.0	0.2456640 04
ROW	6	0.1373690 06	0.2966710 08	-0.4645520 07	0.3801120 02
ROW	7	0.0	-0.4645520 07	-0.3693300 08	0.0
ROW	8	0.2456640 04	0.3801120 02	0.0	0.2428290 07
ROW	9	0.3801120 02	0.1900560 01	-0.4645520 07	0.1195930 06
ROW	10	0.0	-0.2340230 07	-0.9132590 07	0.0
ROW	11	-0.1561310 07	-0.7798090 05	0.0	-0.3193240 06
ROW	12	-0.7798090 05	-0.1203260 07	0.4645520 07	-0.1414430 05
ROW	13	0.0	0.0	-0.1390020 08	0.0
ROW	14	0.1450050 06	0.7177990 04	0.0	-0.1519420 07
ROW	15	0.7177990 04	0.3569000 03	0.2340230 07	-0.7582200 05
ROW	16	0.0	0.0	0.0	0.0
ROW	17	0.3656290 06	0.1835320 05	0.0	0.3288260 05
ROW	18	0.1835320 05	0.9175620 03	0.0	0.1495030 04
ROW	19	0.0	0.0	0.0	0.0
ROW	20	-0.2781490 05	-0.1455000 04	0.0	0.3536020 06
ROW	21	-0.1455000 04	-0.7275010 02	0.0	0.1781270 05
ROW	22	0.0	0.0	0.0	0.0
ROW	23	-0.4455250 05	-0.2156840 04	0.0	0.1365400 06
ROW	24	-0.2156840 04	-0.1080420 03	0.0	0.6694320 04
ROW	25	0.0	0.0	0.0	0.0
ROW	26	0.4786320 04	0.1788920 03	0.0	-0.3604260 05
ROW	27	0.1788920 03	0.8944610 01	0.0	-0.1677430 04
ROW	28	0.0	0.0	0.0	0.0
ROW	29	0.3194860 04	0.2231670 03	0.0	-0.1805130 05
ROW	30	0.2231670 03	0.1100840 02	0.0	-0.1027270 04

COLUMN ,		9	10	11	12
ROW	1	0.2340230 07	0.0	0.0	0.0
ROW	2	-0.7798090 05	0.0	0.2456640 04	0.3801120 02
ROW	3	-0.1203260 07	0.0	0.3801120 02	0.1900560 01
ROW	4	-0.4645520 07	-0.1390020 08	0.0	0.2305300 07
ROW	5	0.3801120 02	0.0	-0.1561310 07	-0.7798090 05
ROW	6	0.1900560 01	0.2340230 07	-0.7798090 05	-0.1203260 07
ROW	7	-0.4645520 07	-0.9132590 07	0.0	0.4645520 07
ROW	8	0.1195930 06	0.0	-0.3193240 06	-0.1414430 05
ROW	9	0.2966620 08	0.4645520 07	-0.1414430 05	-0.2726220 08
ROW	10	0.4645520 07	0.3693300 08	0.0	-0.4645520 07
ROW	11	-0.1414430 05	0.0	0.2428290 07	0.1195930 06
ROW	12	-0.2726220 08	-0.4645520 07	0.1195930 06	0.2966620 08
ROW	13	0.2305300 07	0.0	0.0	-0.4645520 07
ROW	14	-0.7582200 05	0.0	0.3288260 05	0.1495030 04
ROW	15	-0.1203160 07	-0.4645520 07	0.1495030 04	0.7475170 02
ROW	16	0.0	-0.1390020 08	0.0	0.2340230 07
ROW	17	0.1495030 04	0.0	-0.1519420 07	-0.7582200 05
ROW	18	0.7475170 02	0.2305300 07	-0.7582200 05	-0.1203160 07
ROW	19	0.0	0.0	0.0	0.0
ROW	20	0.1781270 05	0.0	0.1365400 06	0.6694320 04
ROW	21	0.8906370 03	0.0	0.6694320 04	0.3347160 03
ROW	22	0.0	0.0	0.0	0.0
ROW	23	0.6694320 04	0.0	0.3536020 06	0.1781270 05
ROW	24	0.3347160 03	0.0	0.1781270 05	0.8906370 03
ROW	25	0.0	0.0	0.0	0.0
ROW	26	-0.1677430 04	0.0	-0.1805130 05	-0.1027270 04
ROW	27	-0.8387150 02	0.0	-0.1027270 04	-0.5136330 02
ROW	28	0.0	0.0	0.0	0.0
ROW	29	-0.1027270 04	0.0	-0.3604260 05	-0.1677430 04
ROW	30	-0.5136330 02	0.0	-0.1677430 04	-0.8387150 02

SCTAMRA - JACKSON - JOANNE USA MATRIX METHOD OF STRUCTURAL ANALYSIS

MATRIX 1 30 ROWS 30 COLUMNS STORAGE MODE 1 PAGE 4

COLUMN	13	14	15	16
ROW 1	0.0	0.0	0.0	0.0
ROW 2	0.0	0.365620D 06	0.183532D 05	0.0
ROW 3	0.0	0.183532D 05	0.917662D 03	0.0
ROW 4	0.0	0.0	0.0	0.0
ROW 5	0.0	0.145005D 06	0.717799D 04	0.0
ROW 6	0.0	0.717799D 04	0.358900D 03	0.0
ROW 7	-0.139002D 08	0.0	0.234023D 07	0.0
ROW 8	0.0	-0.151942D 07	-0.758220D 05	0.0
ROW 9	0.230530D 07	-0.758220D 05	-0.120316D 07	0.0
ROW 10	0.0	0.0	-0.464552D 07	-0.139002D 08
ROW 11	0.0	0.328826D 05	0.149503D 04	0.0
ROW 12	-0.464552D 07	0.149503D 04	0.747517D 02	0.234023D 07
ROW 13	0.369330D 08	0.0	-0.464552D 07	-0.913259D 07
ROW 14	0.0	0.241627D 07	0.119052D 06	0.0
ROW 15	-0.464552D 07	0.119052D 06	0.296651D 08	0.464552D 07
ROW 16	-0.913259D 07	0.0	0.464552D 07	0.369330D 08
ROW 17	0.0	-0.327787D 06	-0.146279D 05	0.0
ROW 18	0.464552D 07	-0.146279D 05	-0.272622D 08	-0.464552D 07
ROW 19	-0.139002D 08	0.0	0.230530D 07	0.0
ROW 20	0.0	-0.148247D 07	-0.739138D 05	0.0
ROW 21	0.234023D 07	-0.739138D 05	-0.120306D 07	-0.464552D 07
ROW 22	0.0	0.0	0.0	-0.139002D 08
ROW 23	0.0	0.682451D 05	0.320263D 04	0.0
ROW 24	0.0	0.320263D 04	0.160131D 03	0.230530D 07
ROW 25	0.0	0.0	0.0	0.0
ROW 26	0.0	0.285865D 06	0.144903D 05	0.0
ROW 27	0.0	0.144903D 05	0.724515D 03	0.0
ROW 28	0.0	0.0	0.0	0.0
ROW 29	0.0	0.981070D 05	0.420830D 04	0.0
ROW 30	0.0	0.420830D 04	0.210415D 03	0.0

COLUMN	17	18	19	20
ROW 1	0.0	0.0	0.0	0.0
ROW 2	0.145005D 06	0.717799D 04	0.0	-0.445025D 05
ROW 3	0.717799D 04	0.358900D 03	0.0	-0.216084D 04
ROW 4	0.0	0.0	0.0	0.0
ROW 5	0.365620D 06	0.183532D 05	0.0	-0.278144D 05
ROW 6	0.183532D 05	0.917662D 03	0.0	-0.145500D 04
ROW 7	0.0	0.0	0.0	0.0
ROW 8	0.328826D 05	0.149503D 04	0.0	0.353602D 06
ROW 9	0.149503D 04	0.747517D 02	0.0	0.178127D 05
ROW 10	0.0	0.230530D 07	0.0	0.0
ROW 11	-0.151942D 07	-0.758220D 05	0.0	0.136540D 06
ROW 12	-0.758220D 05	-0.120316D 07	0.0	0.669032D 04
ROW 13	0.0	0.464552D 07	-0.139002D 08	0.0
ROW 14	-0.327787D 06	-0.146279D 05	0.0	-0.148247D 07
ROW 15	-0.146279D 05	-0.272622D 08	0.230530D 07	-0.739138D 05
ROW 16	0.0	-0.464552D 07	0.0	0.0
ROW 17	0.241627D 07	0.119052D 06	0.0	0.682451D 05
ROW 18	0.119052D 06	0.296651D 08	-0.464552D 07	0.320263D 04
ROW 19	0.0	-0.464552D 07	0.369330D 08	0.0
ROW 20	0.682451D 05	0.320263D 04	0.0	0.213662D 07
ROW 21	0.320263D 04	0.160131D 03	-0.464552D 07	0.105234D 06
ROW 22	0.0	0.234023D 07	-0.913259D 07	0.0
ROW 23	-0.148247D 07	-0.739138D 05	0.0	-0.540277D 06
ROW 24	-0.739138D 05	-0.120306D 07	0.464552D 07	-0.253168D 05
ROW 25	0.0	0.0	-0.139002D 08	0.0
ROW 26	0.981070D 05	0.420830D 04	0.0	-0.950646D 06
ROW 27	0.420830D 04	0.210415D 03	0.234023D 07	-0.472503D 05
ROW 28	0.0	0.0	0.0	0.0
ROW 29	0.285865D 06	0.144903D 05	0.0	0.338461D 06
ROW 30	0.144903D 05	0.724515D 03	0.0	0.166411D 05

MATRIX 1 30 ROWS 30 COLUMNS STORAGE MODE 1

COLUMN	21	22	23	24
ROW 1	0.0	0.0	0.0	0.0
ROW 2	-0.2160850 04	0.0	-0.2781440 03	-0.1455000 04
ROW 3	-0.1080420 03	0.0	-0.1455000 04	-0.7275010 02
ROW 4	0.0	0.0	0.0	0.0
ROW 5	-0.1455450 04	0.0	-0.4350250 05	-0.2160840 04
ROW 6	-0.7275410 02	0.0	-0.2160840 04	-0.1080420 03
ROW 7	0.0	0.0	0.0	0.0
ROW 8	0.1781270 05	0.0	0.1365400 06	0.6694320 04
ROW 9	0.6900320 03	0.0	0.6694320 04	0.3347160 03
ROW 10	0.0	0.0	0.0	0.0
ROW 11	0.6694520 04	0.0	0.3530020 06	0.1781270 05
ROW 12	0.3347160 03	0.0	0.1781270 05	0.6906370 03
ROW 13	0.2380250 07	0.0	0.0	0.0
ROW 14	-0.7391360 05	0.0	0.6824510 05	0.3202630 04
ROW 15	-0.1293060 07	0.0	0.3202630 04	0.1601310 03
ROW 16	-0.4645520 07	-0.1390020 08	0.0	0.2305300 07
ROW 17	0.3202630 04	0.0	-0.1482470 07	-0.7391380 05
ROW 18	0.1601310 03	0.2366230 07	-0.7391380 05	-0.1293060 07
ROW 19	-0.4645520 07	-0.4152590 07	0.0	0.4645520 07
ROW 20	0.1052340 06	0.0	-0.5402770 06	-0.2531680 05
ROW 21	0.2966540 06	0.4645520 07	-0.2531680 05	-0.2726270 08
ROW 22	[illegible]	-0.7653600 06	0.0	-0.4645520 07
ROW 23	-0.2531680 05	0.0	0.2138620 07	0.1052340 06
ROW 24	-0.2726270 08	-0.4645520 07	0.1052340 06	0.2966540 06
ROW 25	0.2305300 07	0.0	0.0	-0.4645520 07
ROW 26	-0.4725030 05	0.0	0.3384610 06	0.1664110 05
ROW 27	-0.1201730 07	-0.4645520 07	0.1664110 05	0.8320550 03
ROW 28	0.0	-0.1390020 08	0.0	0.2340230 07
ROW 29	0.1664110 05	0.0	-0.9506460 06	-0.4725030 05
ROW 30	0.8320550 03	0.2305300 07	-0.4725030 05	-0.1201730 07

COLUMN	25	26	27	28
ROW 1	0.0	0.0	0.0	0.0
ROW 2	0.0	0.3194860 04	0.2201670 03	0.0
ROW 3	0.0	0.2201670 03	0.1100840 02	0.0
ROW 4	0.0	0.0	0.0	0.0
ROW 5	0.0	0.4786320 04	0.1788920 03	0.0
ROW 6	0.0	0.1788920 03	0.8944610 01	0.0
ROW 7	0.0	0.0	0.0	0.0
ROW 8	0.0	-0.3605260 05	-0.1677430 04	0.0
ROW 9	0.0	-0.1677430 04	-0.8387150 02	0.0
ROW 10	0.0	0.0	0.0	0.0
ROW 11	0.0	-0.1465530 05	-0.1027270 04	0.0
ROW 12	0.0	-0.1027270 04	-0.5136330 02	0.0
ROW 13	0.0	0.0	0.0	0.0
ROW 14	0.0	0.2656630 06	0.1429030 05	0.0
ROW 15	0.0	0.1449030 05	0.7245150 03	0.0
ROW 16	0.0	0.0	0.0	0.0
ROW 17	0.0	0.6810740 05	0.4208300 04	0.0
ROW 18	0.0	0.4208300 04	0.2104150 03	0.0
ROW 19	-0.1390020 08	0.0	0.2340230 07	0.0
ROW 20	0.0	-0.9506460 06	-0.4725030 05	0.0
ROW 21	0.2305300 07	-0.4725030 05	-0.1201730 07	0.0
ROW 22	0.0	0.0	-0.4645520 07	-0.1390020-08
ROW 23	0.0	0.3384610 06	0.1664110 05	0.0
ROW 24	-0.4645520 07	0.1664110 05	0.8320530 03	0.2340230 07
ROW 25	-0.1846650-08	0.0	0.0	-0.4566290-07
ROW 26	0.0	0.7171780 06	0.3424680 05	0.0
ROW 27	0.0	0.3424680 05	0.1483180 08	0.2305300 07
ROW 28	-0.4566290-07	0.0	0.2305300 07	0.1846650-08
ROW 29	0.0	-0.4317280 06	-0.1997400 05	0.0
ROW 30	0.2340230 07	-0.1997400 05	-0.1363170 08	-0.4645520 07

MATRIX 1 30 ROWS 30 COLUMNS STORAGE MODE 1

	COLUMN	29	30
ROW	1	0.0	0.0
ROW	2	0.3786320 04	0.1788920 03
ROW	3	0.1788920 03	0.8944610 01
ROW	4	0.0	0.0
ROW	5	0.3194660 04	0.2201670 03
ROW	6	0.2201670 03	0.1100840 02
ROW	7	0.0	0.0
ROW	8	-0.1605130 05	-0.1027270 04
ROW	9	-0.1027270 04	-0.5136330 02
ROW	10	0.0	0.0
ROW	11	-0.3654260 05	-0.1677430 04
ROW	12	-0.1677430 04	-0.8387150 02
ROW	13	0.0	0.0
ROW	14	0.8610700 05	0.4208300 04
ROW	15	0.4208300 04	0.2104150 03
ROW	16	0.0	0.0
ROW	17	0.2958660 06	0.1449030 05
ROW	18	0.1449030 05	0.7245150 03
ROW	19	0.0	0.0
ROW	20	0.3354610 05	0.1664110 05
ROW	21	0.1664110 05	0.8320530 03
ROW	22	0.0	0.2305300 07
ROW	23	-0.9560630 06	-0.4725630 05
ROW	24	-0.4725630 05	-0.1263730 07
ROW	25	0.0	0.2349250 07
ROW	26	-0.4517230 06	-0.1997400 05
ROW	27	-0.1997400 05	-0.1365170 06
ROW	28	0.0	-0.4605920 07
ROW	29	0.7171700 06	0.3424680 05
ROW	30	0.3424680 05	0.1483180 06

MASSES

3	0.13000 00	0.13000 00	0.13000 00	0.0	0.0	0.0
4	0.13000 00	0.13000 00	0.13000 00	0.0	0.0	0.0
5	0.13000 00	0.13000 00	0.13000 00	0.0	0.0	0.0
6	0.13000 00	0.13000 00	0.13000 00	0.0	0.0	0.0
7	0.13000 00	0.13000 00	0.13000 00	0.0	0.0	0.0
8	0.13000 00	0.13000 00	0.13000 00	0.0	0.0	0.0
9	0.13000 00	0.13000 00	0.13000 00	0.0	0.0	0.0
10	0.13000 00	0.13000 00	0.13000 00	0.0	0.0	0.0
11	0.13000 00	0.13000 00	0.13000 00	0.0	0.0	0.0
12	0.13000 00	0.13000 00	0.13000 00	0.0	0.0	0.0

EIGENVALUES

0.15[illegible] 03	RAD/SEC	EIGENVALUE =	0.22773519D 05	CPS =	0.2401800 02
0.332[illegible] 03	RAD/SEC	EIGENVALUE =	0.11041[illegible]D 06	CPS =	0.5288560 02
0.93[illegible] 03	RAD/SEC	EIGENVALUE =	0.876[illegible] 06	CPS =	0.1391360 03
0.94[illegible] 03	RAD/SEC	EIGENVALUE =	0.69150960D 06	CPS =	0.1502740 03
0.15538127 04	RAD/SEC	EIGENVALUE =	0.24143340D 07	CPS =	0.2472980 03
0.23861[illegible] 04	RAD/SEC	EIGENVALUE =	0.56935562D 07	CPS =	0.3797670 03
0.25[illegible] 04	RAD/SEC	EIGENVALUE =	0.63[illegible] 07	CPS =	0.4014910 03
0.2[illegible] 04	RAD/SEC	EIGENVALUE =	0.80622244D 07	CPS =	0.4629820 03
0.32[illegible] 04	RAD/SEC	EIGENVALUE =	0.10781863D 08	CPS =	0.5225990 03
0.39[illegible] 04	RAD/SEC	EIGENVALUE =	0.15492358D 08	CPS =	0.6264460 03
0.44778[illegible] 04	RAD/SEC	EIGENVALUE =	0.20051271D 08	CPS =	0.7126770 03
0.47[illegible] 04	RAD/SEC	EIGENVALUE =	0.22945159D 08	CPS =	0.7623730 03
0.54[illegible] 04	RAD/SEC	EIGENVALUE =	0.29514031D 08	CPS =	0.8646420 03
0.56[illegible] 04	RAD/SEC	EIGENVALUE =	0.32347784D 08	CPS =	0.9057590 03
0.62935[illegible] 04	RAD/SEC	EIGENVALUE =	0.39612747D 08	CPS =	0.1001650 04
0.65528[illegible] 04	RAD/SEC	EIGENVALUE =	0.42939514D 08	CPS =	0.1042920 04
0.84[illegible] 04	RAD/SEC	EIGENVALUE =	0.71916608D 08	CPS =	0.1349700 04
0.10868[illegible] 05	RAD/SEC	EIGENVALUE =	0.11813165D 09	CPS =	0.1729840 04
0.131[illegible] 05	RAD/SEC	EIGENVALUE =	0.17278440D 09	CPS =	0.2092060 04
0.138[illegible] 05	RAD/SEC	EIGENVALUE =	0.19200285D 09	CPS =	0.2205340 04
0.14788[illegible] 05	RAD/SEC	EIGENVALUE =	0.21872657D 09	CPS =	0.2353720 04
0.17237[illegible] 05	RAD/SEC	EIGENVALUE =	0.29711443D 09	CPS =	0.2743370 04
0.174[illegible] 05	RAD/SEC	EIGENVALUE =	0.30309523D 09	CPS =	0.2770840 04
0.186[illegible] 05	RAD/SEC	EIGENVALUE =	0.34852554D 09	CPS =	0.2971250 04
0.190[illegible] 05	RAD/SEC	EIGENVALUE =	0.36252107D 09	CPS =	0.3030320 04
0.19872[illegible] 05	RAD/SEC	EIGENVALUE =	0.39491957D 09	CPS =	0.3162830 04
0.2063[illegible] 05	RAD/SEC	EIGENVALUE =	0.42581779D 09	CPS =	0.3284230 04
0.21380[illegible] 05	RAD/SEC	EIGENVALUE =	0.45713210D 09	CPS =	0.3402850 04
0.23253677 05	RAD/SEC	EIGENVALUE =	0.54073350D 09	CPS =	0.3700950 04
0.25100[illegible] 05	RAD/SEC	EIGENVALUE =	0.63005494D 09	CPS =	0.3994950 04

TIME = 0.0131460-02

EIGENVALUES

	0.22773520 05	0.11041640 06	0.87605730 06

EIGENVECTORS

ROW	MODE			
1	3	-0.1785[illegible]-03	-0.13811590-01	0.57705520-03
2	3	-0.33196[illegible]-01	-0.87417700-04	0.17131140 00
3	3	0.54868040-03	0.42561730-01	-0.62616500-02
4	4	0.1[illegible]-03	0.1392[illegible]80-01	-0.61283310-03
5	4	-0.33196640-01	-0.11869760-03	0.17129740 00
6	4	0.55752670-03	0.43250940-01	-0.64601690-02
7	5	-0.30435[illegible]-03	-0.23516520-01	-0.32952530-04
8	5	-0.11899[illegible] 00	0.61327610-03	0.39890650 00
9	5	0.10[illegible]220-02	0.13119330 00	-0.11271990-01
10	6	0.3[illegible]7440-03	0.23657880-01	-0.55226830-04
11	6	-0.11[illegible] 00	0.78652840-03	0.39885830 00
12	6	0.17[illegible]70-02	0.13154270 00	-0.11373970 01
13	7	-0.[illegible]-03	-0.29521330-01	-0.12069450-02
14	7	-0.2[illegible] 00	0.25125440-02	0.40617930 00
15	7	0.3[illegible]-02	0.24969980 00	-0.99135940-02
16	8	0.34955870-03	0.30763510-01	0.11068140-02
17	8	-0.21025280 00	0.24919410-02	0.40609870 00
18	8	0.52322630-02	0.24989860 00	-0.98746530-02
19	9	-0.4[illegible]5520-03	-0.32[illegible]-01	-0.21929520-02
20	9	-0.[illegible] 00	0.47903980-02	0.11297370 00
21	9	0.4[illegible]-02	0.38225260 00	-0.14134710-02
22	10	-0.4[illegible]-03	0.33107170-01	0.21126810-02
23	10	-0.38147540 00	0.47783040-02	0.11286710 00
24	10	0.47535300-02	0.38263480 00	-0.13046290-02
25	11	-0.4295[illegible]-03	-0.33152350-01	-0.25362720-02
26	11	-0.55047970 00	0.74573870-02	-0.36520030 00
27	11	0.6[illegible]-02	0.51789190 00	0.10427060-01
28	12	0.45[illegible]-03	0.33962160-01	0.24966180-02
29	12	-0.55[illegible] 00	0.74499870-02	-0.36537500 00
30	12	0.67112640-02	0.51817810 00	0.10554360-01

EIGENVECTORS NORMALIZED WITH RESPECT TO THE MAXIMUM ELEMENT

ROW	NODE			
1	3	0.33655490-03	-0.26654140-01	0.14206860-02
2	3	0.62577780-01	-0.16870200-03	0.42176290 00
3	3	-0.10343100-02	0.82137260-01	-0.15415980-01
4	4	-0.33933340-03	0.26879720-01	-0.15087750-02
5	4	0.62578520-01	-0.22906720-03	0.42172950 00
6	4	-0.10509860-02	0.83467320-01	-0.15904720-01
7	5	0.57370670-03	-0.45383080-01	-0.81128040-04
8	5	0.22432240 00	0.15694910-02	0.98209460 00
9	5	-0.31945070-02	0.25318200 00	-0.27751270-01
10	6	-0.58188160-03	0.46041850-01	-0.13596660-03
11	6	0.22432300 00	0.15178730-02	0.98197610 00
12	6	-0.32102960-02	0.25443510 00	-0.28002350-01
13	7	0.72083520-03	-0.56971390-01	-0.29763830-02
14	7	0.45289570 00	0.48488040-02	0.10000000 01
15	7	-0.60796690-02	0.48120490 00	-0.24382320-01
16	8	-0.73386540-03	0.58017710-01	0.27249380-02
17	8	0.45289720 00	0.48090440-02	0.99980150 00
18	8	-0.60536930-02	0.48226390 00	-0.24311070-01
19	9	0.79145170-03	-0.62544220-01	-0.53990000-02
20	9	0.71911360 00	0.92446950-02	0.27813010 00
21	9	-0.93284360-02	0.73766590 00	-0.34799180-02
22	10	-0.80847360-03	0.63891490-01	0.52013520-02
23	10	0.71911400 00	0.92213550-02	0.27787520 00
24	10	-0.93378300-02	0.73642340 00	-0.32119530-02
25	11	0.80971660-03	-0.63978680-01	-0.62491420-02
26	11	0.99999990 00	0.14391550-01	-0.89925670 00
27	11	-0.12644390-01	0.99944770 00	0.25671070-01
28	12	-0.62925400-03	0.65541480-01	0.61470630-02
29	12	0.10000000 01	0.14377270-01	-0.89954120 00
30	12	-0.12651350-01	0.10000000 01	0.25984530-01

EIGENVALUES

0.89150960 06 0.24143340 07

EIGENVECTORS

ROW	NODE		
1	3	0.28715350-05	0.21402940-01
2	3	0.95859410-01	0.16509620-03
3	3	-0.14050300-04	-0.23561440 00
4	4	0.77073890-06	-0.22977480-01
5	4	-0.95903580-01	0.67292730-03
6	4	0.15911410-06	-0.24298430 00
7	5	0.68537260-05	-0.99679570-03
8	5	0.21539660 00	-0.64359960-02
9	5	-0.29665450-04	-0.41586540 00
10	6	0.53254600-05	-0.27041130-02
11	6	-0.21550170 00	-0.60556990-02
12	6	0.30379560-07	-0.41940720 00
13	7	0.12494300-04	-0.42738400-01
14	7	0.31859750 00	-0.13554080-01
15	7	-0.40566090-04	-0.36489450 00
16	8	0.12895270-04	0.38405220-01
17	8	-0.31570240 00	-0.13447690-01
18	8	0.24342780-05	-0.36375940 00
19	9	0.19778050-04	-0.76949230-01
20	9	0.39352510 00	-0.61905880-02
21	9	-0.44133920-04	-0.69492930-01
22	10	0.22489900-04	0.73293970-01
23	10	-0.39334870 00	-0.62933920-02
24	10	0.69555460-05	-0.65724550-01
25	11	0.26160520-04	-0.89102810-01
26	11	0.43379660 00	0.12531710-01
27	11	-0.54383810-04	0.33560600 00
28	12	0.44562140-04	0.86765730-01
29	12	-0.43370210 00	0.12336860-01
30	12	0.59543930-04	0.34138400 00

SCIARRA - JACKSON - JOANNE USA METHOD OF STRUCTURAL ANALYSIS

EIGENVECTORS NORMALIZED WITH RESPECT TO THE MAXIMUM ELEMENT

ROW	NODE		
1	3	0.5619551D-05	-0.5103140D-01
2	3	0.2209781D 00	-0.3936417D-03
3	3	-0.3238919D-04	0.5617795D 00
4	4	0.1776731D-05	0.5478561D-01
5	4	-0.2210799D 00	-0.1604473D-02
6	4	0.3667997D-06	0.5793518D 00
7	5	0.1579942D-04	0.2376678D-02
8	5	0.4965435D 00	0.2011410D-01
9	5	-0.6838658D-04	0.9915554D 00
10	6	0.1227642D-03	0.6447465D-02
11	6	-0.4467812D 00	0.1920735D-01
12	6	0.7003190D-07	0.1000000D 01
13	7	0.2880225D-04	0.1019019D 00
14	7	0.7344409D 00	0.3231723D-01
15	7	-0.9551628D-04	0.8700246D 00
16	8	0.2972654D-04	-0.9157024D-01
17	8	-0.7346626D 00	0.3206356D-01
18	8	0.5611573D-05	0.8673181D 00
19	9	0.4559298D-04	0.1834714D 00
20	9	0.9066936D 00	0.1476033D-01
21	9	-0.1017389D-03	0.1656932D 00
22	10	0.5188442D-04	-0.1747561D 00
23	10	-0.9067596D 00	0.1500545D-01
24	10	0.2064860D-04	0.1567082D 00
25	11	0.6030605D-04	0.2124494D 00
26	11	0.1000000D 01	-0.2987957D-01
27	11	-0.1253673D-03	-0.8032434D 00
28	12	0.1027260D-03	-0.2068771D 00
29	12	-0.9997835D 00	-0.2941500D-01
30	12	0.1395677D-03	-0.8139679D 00

SCIARRA - JACKSON

(DA2 PROGRAM)

DAMPED FORCED RESPONSE OF COMPLEX STRUCTUR,

INPUT

NUMBER OF ROWS = 30

NUMBER OF COLUMNS = 5

TAPE/CARD INPUT OPTION =****

EIGENVECTORS DESIRED

1

2

3

4

5

EIGENVALUES	MODAL DAMPING
0.2277350 05	0.200000D-01
0.1104160 06	0.200000D-01
0.8760570 06	0.200000D-01
0.8915100 06	0.200000D-01
0.2414330 07	0.200000D-01

NATURAL FREQ RAD/SEC

0.1509090 03

0.3322900 03

0.9370470 03

0.9441980 03

0.1553810 04

SCIARRA - JACKSON
(DR2 PROGRAM)
DAMPED FORCED RESPONSE OF COMPLEX STRUCTURE

MODAL MATRIX

COLUMNS / ROW	1	2	3	4	5
1	-0,17855556D-03	-0,13611589D-01	0,57705320D-03	0,28715347D-05	0,21402936D-01
2	-0,33196244D-01	-0,87417701D-04	0,17131135D 00	0,95659406D-01	0,16509015D-03
3	0,54068041D-03	-0,42561728D-01	-0,62616505D-02	-0,14050303D-04	-0,23501436D 00
4	0,18000953D-03	0,13928482D-01	-0,61283306D-03	0,77073888D-06	-0,22977476D-01
5	-0,33196636D-01	-0,11869759D-03	0,17129776D 00	-0,95903576D-01	0,67292727D-03
6	0,55752670D-03	0,43250938D-01	-0,64601688D-02	0,15911409D-06	-0,24298427D 00
7	-0,30433981D-03	-0,23510518D-01	-0,32952530D-04	0,68537257D-05	-0,99679571D-03
8	-0,11899648D 00	0,81327609D-05	0,39890606D 00	0,21539869D 00	-0,64359960D-02
9	0,16946215D-02	-0,13119354D 00	-0,11271991D-01	-0,29665826D-04	-0,41586544D 00
10	0,30867642D-03	0,23657877D-01	-0,55226835D-04	0,54254602D-05	-0,27041131D-02
11	-0,11899881D 00	0,78652845D-03	0,39885833D 00	-0,21550171D 00	-0,80556989D-02
12	0,17029979D-02	-0,13184270D 00	-0,11373974D-01	0,30379557D-07	-0,41940716D 00
13	-0,35238848D-03	-0,29521326D-01	-0,12089456D-02	0,12494303D-04	-0,42738395D-01
14	-0,24025253D 00	0,25125437D-02	0,40617927D 00	0,31859752D 00	-0,13554079D-01
15	0,32251413D-02	-0,24934984D 00	-0,99035936D-02	-0,40566989D-04	-0,36469455D 00
16	0,30930075D-03	0,30063507D-01	0,11068135D-02	0,12895272D-04	0,38405216D-01
17	-0,24025278D 00	0,24919414D-02	0,40609865D 00	-0,31870237D 00	-0,13447688D-01
18	0,32327626D-02	-0,24999861D 00	-0,98746531D-02	0,24342781D-05	-0,36375940D 00
19	-0,41995516D-03	-0,32409042D-01	-0,21929619D-02	0,19778054D-04	-0,76949225D-01
20	-0,38147530D 00	0,47963962D-02	0,11297069D 00	0,39352006D 00	-0,61905878D-02
21	0,49965475D-02	-0,38225265D 00	-0,14134705D-02	-0,44133918D-04	-0,69492925D-01
22	0,42867586D-03	0,33107172D-01	0,21126615D-02	0,22489904D-04	0,73293972D-01
23	-0,38147543D 00	0,47783039D-02	0,11286715D 00	-0,39334870D 00	-0,62933915D-02
24	0,49553295D-02	-0,38263481D 00	-0,13048265D-02	0,69555456D-05	-0,65724548D-01
25	-0,42753625D-03	-0,33152351D-01	-0,25362719D-02	0,20169525D-04	-0,69102813D-01
26	-0,53047969D 00	0,74573572D-02	-0,36526026D 00	0,43379600D 00	0,12531706D-01
27	0,67075935D-02	-0,51784191D 00	-0,10427058D-01	-0,54383815D-04	0,33688603D 00
28	0,43990247D-03	0,33962159D-01	0,24968176D-02	0,44562140D-04	0,86765730D-01
29	-0,53047976D 00	0,74499667D-02	-0,36537501D 00	-0,43370208D 00	0,12336863D-01
30	0,67112840D-02	-0,51817809D 00	-0,10554378D-01	0,60543927D-04	-0,34138399D 00

FINAL EFFECTIVE MASS MATRIX

COLUMNS / ROW	1	2	3	4	5
1	0,13000000D 00	-0,10298018D-12	0,26870744D-15	0,25445615D-13	-0,25208785D-14
2	-0,10298034D-12	0,13000000D 00	0,53639439D-13	0,21931996D-11	-0,36496582D-14
3	-0,27560654D-15	-0,53639601D-13	0,13000000D 00	-0,26003849D-11	-0,17821898D-13
4	0,25444748D-13	0,21931996D-11	0,26003849D-11	0,13000000D 00	-0,74113194D-12
5	-0,25210953D-14	-0,36463887D-14	-0,17821896D-13	-0,74113194D-12	0,13000000D 00

OSCILLATORY FORCES FOR THIS CASE

FORCE COMPONENTS

ROW	SINE	COSINE
27	0,1000000D 04	0,1000000D 04

PSEUDO-VALUES

M EFF	C	K	F EFF SINE	F EFF COS
0,1300000 00	0,784727D 00	0,296056D 04	0,670759D 01	0,670759D 01
0,130000D 00	0,172791D 01	0,143554D 05	0,517892D 03	0,517892D 03
0,130000D 00	0,487265D 01	0,114147D 06	0,104271D 02	0,104271D 02
0,130000D 00	0,490983D 01	0,115896D 06	-0,543838D-01	-0,543838D-01
0,130000D 00	0,807983D 01	0,313863D 06	0,336886D 03	0,336886D 03

SCIARRA - JACKSON
(DR2 PROGRAM)
DAMPED FORCED RESPONSE OF COMPLEX STRUCTUR.

EXCITING FREQUENCY = 62,83180 RAD/SEC

SINE COMPONENT
PSEUDO-VALUES

FORCING FREQ.	AMPLIFICATION FACTOR	PHASE ANGLE(RAD)	AMPLITUDE
0,416356	1,209459	0,020144	0,002740
0,189087	1,037048	0,007844	0,037416
0,067053	1,004513	0,002694	0,000092
0,066545	1,004444	0,002674	-0,000000
0,040437	1,001637	0,001620	0,001075

COSINE COMPONENT
PSEUDO-VALUES

FORCING FREQ.	AMPLIFICATION FACTOR	PHASE ANGLE(RAD)	AMPLITUDE
0,416356	1,209459	0,020144	0,002740
0,189087	1,037048	0,007844	0,037416
0,067053	1,004513	0,002694	0,000092
0,066545	1,004444	0,002674	-0,000000
0,040437	1,001637	0,001620	0,001075

NODE	SINE COMPONENT	COSINE COMPONENT	RESULTANT	PHASE ANGLE	G LOADING
3	0,000490	-0,000498	0,000699	135,47	0,00715
3	0,000077	-0,000080	0,000111	136,33	0,00113
3	-0,001328	0,001352	0,001895	315,52	0,01938
4	-0,000493	0,000501	0,000703	315,47	0,00719
4	0,000077	-0,000081	0,000112	136,32	0,00114
4	-0,001346	0,001370	0,001920	315,52	0,01964
5	0,000875	-0,000889	0,001247	135,45	0,01275
5	0,000262	-0,000274	0,000379	136,33	0,00388
5	-0,004427	0,004503	0,006315	315,49	0,06459
6	-0,000884	0,000898	0,001260	315,45	0,01288
6	0,000262	-0,000275	0,000380	136,33	0,00389
6	-0,004468	0,004524	0,006344	315,49	0,06488
7	0,001143	-0,001160	0,001629	135,44	0,01666
7	0,000529	-0,000554	0,000766	136,32	0,00784
7	-0,008872	0,009018	0,012651	315,47	0,12939
8	-0,001158	0,001176	0,001651	315,44	0,01688
8	0,000530	-0,000554	0,000767	136,32	0,00784
8	-0,008894	0,009040	0,012681	315,47	0,12970
9	0,001287	-0,001306	0,001834	135,43	0,01876
9	0,000843	-0,000882	0,001220	136,30	0,01248
9	-0,014126	0,014353	0,020140	315,45	0,20598
10	-0,001309	0,001329	0,001865	315,43	0,01908
10	0,000843	-0,000882	0,001220	136,30	0,01248
10	-0,014147	0,014372	0,020166	315,45	0,20625
11	0,001328	-0,001348	0,001892	135,42	0,01935
11	0,001167	-0,001222	0,001690	136,30	0,01728
11	-0,019606	0,019911	0,027944	315,44	0,28579
12	-0,001355	0,001376	0,001931	315,43	0,01975
12	0,001167	-0,001222	0,001690	136,30	0,01728
12	-0,019621	0,019927	0,027966	315,44	0,28602

SCIARRA/MOFFA

VIBRATORY INTERNAL LOADS

CASE 1

SINE DEFLECTIONS (INCHES) (A)	SINE LOADS LBS & IN-LBS (AS)	COSINE DEFLECTIONS (INCHES) (B)	COSINE LOADS LBS & IN-LBS (BC)
0,00049016	0,1445	-0,00049821	-0,2374
0,00007655	-1,5295	-0,00008018	1,5277
-0,00132796	56,7582	0,00135218	-56,6574
-0,00049281	0,3241	0,00050093	-0,2318
0,00007768	-1,6942	-0,00008073	1,6928
-0,00134564	58,6758	0,00137024	-58,7775
0,00087456	12,6693	-0,00088870	-13,0687
0,00026191	-0,8008	-0,00027440	0,7820
-0,00442726	70,2990	0,00450300	-69,6484
-0,00088357	-11,8035	0,00089759	12,0017
0,00026230	-0,9376	-0,00027480	0,9191
-0,00444757	71,1471	0,00452368	-70,4950
0,00114289	30,1442	-0,00116041	-30,4399
0,00052921	-0,9113	-0,00055410	-0,9606
-0,00887237	-9,8587	0,00901781	11,3611
-0,00115638	-26,9723	0,00117620	29,2678
0,00052957	-0,6526	-0,00055447	-0,9018
-0,00889396	-10,5339	0,00903975	12,2422
0,00128750	43,2188	-0,00130634	-43,5763
-0,00084272	1,4051	-0,00088199	-1,4909
-0,01412842	-180,2962	0,01435309	183,4424
-0,00130911	-42,3586	0,00132875	42,7180
-0,00084292	1,4054	-0,00088219	-1,4902
-0,01414667	-181,7705	0,01437158	184,9241
0,00132770	47,7130	-0,00134751	-48,0900
-0,00116740	-0,1404	-0,00122172	0,0228
-0,01966557	-389,6143	0,01991148	394,3486
-0,00135527	-47,3568	0,00137556	47,7381
0,00116769	-0,1169	-0,00122181	-0,0007
-0,01962105	-391,2832	0,01992713	396,0251

```
77777
        CH-47C  XMSN SYSTEM COUPLED SCIARRA .5 POST 122 DOF
0001579.            173.            122.            283.            1.              9353
00065599.           29.0009000E 060.                0.              .333            9353
001150.             0.              1.              20.             42.
001651.             1.              0.              0.              0.              9468
002150.             0.              0.              0.              0.              8829
002651.             0.              0.              0.              0.              935
003121.             6.
88888
100-10.0     0.0        0.0        000000SYNC SHAFT
1010.0001    0.0        0.0        022211
1030.40      0.0        0.0        011111
1050.75      0.0        0.0        022211
1071.0       0.0        0.0        011111
1091.4       0.0        0.0        022211
1113.4       0.0        0.0        022211
1133.75      0.0        0.0        011111
1154.7       0.0        0.0        022211
1175.0       0.0        0.0        011111
1195.3       0.0        0.0        011111
1215.6       0.0        0.0        022211
1236.0       0.0        0.0        011111
1256.55      0.0        0.0        022211BRG1
1277.25      0.0        0.0        011111
1287.55      0.0        0.0        022211PROBES
1298.95      0.0        0.0        022211BRG 2
13110.25     0.0        0.0        022211BRG 3
13211.25     0.0        0.0        022211PROBES
13311.75     0.0        0.0        011111
13512.75     0.0        0.0        022211BRG 4
13713.5      0.0        0.0        011111
13913.85     0.0        0.0        022211
14114.1      0.0        0.0        011111
14314.6      0.0        0.0        011111
14515.0      0.0        0.0        022211S.B. PINION
14715.5      0.0        0.0        011111
14915.75     0.0        0.0        011111
15116.00     0.0        0.0        022211
2256.55      10.0       0.1        000000
3256.55      0.1        10.0       000000
23512.75     10.0       0.1        000000
33512.75     0.1        10.0       000000
24515.00     3.2        0.0        222101
24615.00     -3.2       0.0        222101
34515.00     0.0        3.2        222110
34615.00     0.0        -3.2       222110
4016.83      -10.0000   8.8500     022211
4037.54      -10.0000   8.8500     011111
4058.03      -10.0000   8.8500     022211BRG 1
4078.53      -10.0000   8.8500     011111
4099.01      -10.0000   8.8500     022211BRG 2
4119.51      -10.0000   8.8500     011111
4129.71      -10.0000   8.8500     022211PROBES
4139.99      -10.0000   8.8500     011111
41510.7      -10.0000   8.8500     022211
41711.45     -10.0000   8.8500     011111
41911.65     -10.0000   8.8500     022211
42112.7      -10.0000   8.8500     022211
42313.75     -10.0000   8.8500     022211
42514.8      -10.0000   8.8500     011111
42715.00     -10.0000   8.8500     022211S.B. GEAR
42915.3      -10.0000   8.8500     011111
43115.81     -10.0000   8.8500     022211
43316.16     -10.0000   8.8500     011111
```

Figure 40. D-82/C-51 Model of Internal Components.

43516.63	-10.0000	8.8500	022211BRG 5	
43717.090	-10.0000	8.8500	011111	
43917.67	-10.0000	8.8500	022211PROBES	
44118.25	-10.0000	8.8500	011111	
44318.51	-10.0000	8.8500	022211	
44519.41	-10.0000	8.8500	022211SUNGEAR	
44720.3	-10.0000	8.8500	022211	
44929.41	-10.0000	8.8500	000000GND	
5058.03	0.0	8.9500	000000	
58.03	-10.1000	18.8500	000000	
53516.630	0.0	8.95	000000	
3516.630	-10.1000	18.8500	000000	
52715.000	-4.35	8.8500	222101	
52815.000	-15.6500	8.8500	222101	
2715.000	-10.0000	14.5000	222110	
2815.000	-10.0000	3.2000	222110	
54519.4100	-7.1600	8.8500	212101	
54619.4100	-12.8400	8.8500	212101	
54719.4100	2.8400	8.8500	000000	
54819.4100	-22.8400	8.8500	000000	
4519.4100	-10.0000	11.6900	221110	
4619.4100	-10.0000	6.0100	221110	
4719.4100	-10.0000	21.6900	000000	
4819.4100	-10.0000	-3.9900	000000	9
002125225	1.41			
004125225325	.01	63.6	.01	
002125325	.78			
004125325225	.01	89.6	.01	
002135235	3.21			
004135235335	.01	63.6	.01	
002135335	2.03			
004135335235	.01	89.6	.01	
002405505	.77			
004405505005	.01	115.5	.01	
002405005	.94			
004405005407	.01	106.5	.01	
002435535	2.06			
004435535035	.01	115.5	.01	
002435035	1.68			
004435035535	.01	106.5	.01	
004101103225	231.0	462.0	40.9	
004101103325	231.0		40.1	
004103105225	12.4	24.9	7.9	
004103105325	12.4		7.9	
004105107225	2.5	5.0	3.0	
004105107325	2.5		3.0	
004107109225	1.7	3.4	3.3	
004107109325	1.7		3.3	
004109111225	1.7	3.4	3.3	
004109111325	1.7		3.3	
004111113225	0.63	1.3	2.3	
004111113325	0.63		2.3	
004113115225	1.34	2.7	3.5	
004113115325	1.34		3.5	
004115117225	1.8	3.5	2.5	
004115117325	1.8		2.5	
004117119225	11.0	22.0	6.2	
004117119325	11.0		6.2	
004119121225	7.3	14.6	3.6	
004119121325	7.3		3.6	
004121123225	7.3	14.6	3.6	
004121123325	7.3		3.6	
004123125225	6.4	12.8	3.1	
004123125325	6.4		3.1	
004125127225	5.7	11.5	2.7	
004125127325	5.7		2.7	
00412712827	5.7	11.5	2.7	
004127128325	5.7		2.7	
004128129225	5.7	11.5	2.7	

004128129325	5.7		2.7
004129131225	5.7	11.5	2.7
004129131325	5.7		2.7
004131132235	5.7	11.5	2.7
004131132335	5.7		2.7
004132133235	5.7	11.5	2.7
004132133335	5.7		2.7
004133135335	13.0		5.6
004133135235	13.0	26.0	5.6
004135137235	7.57	15.2	2.6
004135137335	7.57		2.6
004137139235	25.6	51.2	9.3
004137139335	25.6		9.3
004139141235	110.7	221.4	24.7
004139141335	110.7		24.7
004141143235	109.3	218.5	22.1
004141143335	109.3		22.1
004143145235	65.4	130.8	14.4
004143145335	65.4		14.4
004145147235	105.9	211.7	22.6
004145147335	105.9		22.6
004147149235	32.1	64.1	9.7
004147149335	32.1		9.7
004149151235	14.2	28.5	4.2
004149151335	14.2		4.2
004100101225	.01	.085	.01
004145245345	15.6	130.	22.6
004145245135	113.5		22.6
002145245	4.16		
004145246345	15.6	130.	22.6
004145246135	113.5		22.6
002145246	4.16		
004145345245	15.6	130.	22.6
004145345135	113.5		22.6
002145345	4.16		
004145346245	15.6	130.	22.6
004145346135	113.5		22.6
002145346	4.16		
002245345	2.08		
002245346	2.08		
002246345	2.08		
002246346	2.08		
002345028	2.92		
004401403505	2.2	4.5	2.6
004401403005	2.2		2.6
004403405505	2.8	5.5	3.1
004403405005	2.8		3.1
004405407505	2.8	5.5	3.1
004405407005	2.8		3.1
004407409505	2.8	5.5	3.1
004407409005	2.8		3.1
004409411505	2.8	5.5	3.1
004409411005	2.8		3.1
004411412505	4.4	8.7	3.9
004411412005	4.4		3.9
004412413505	3.2	6.4	2.6
004412413005	3.2		2.6
004413415505	3.6	7.2	2.6
004413415005	3.6		2.6
004415417505	4.1	8.3	2.9
004415417005	4.1		2.9
004417419505	8.2	16.3	5.1
004417419005	8.2		5.1
0044194215	[illegible]	12.5	4.1
004419421005	6.2		4.1
004421423505	10.5	21.1	5.6
004421423005	10.5		5.6
004423425505	14.3	28.6	6.1
004423425005	14.3		6.1

004425427505	14.5	29.0	4.9
004425427005	14.5		4.9
004427429535	199.6	399.1	36.1
004427429035	199.6		36.1
004429431535	31.5	63.0	7.8
004429431035	31.5		7.8
004431433535	21.1	42.1	5.0
004431433535	21.1		5.0
004433435535	42.9	85.8	9.3
004433435035	42.9		9.3
004435437535	42.9	85.8	9.3
004435437035	42.9		9.3
004437439535	16.3	32.6	4.3
004437439035	16.3		4.3
004439441535	11.2	22.5	3.8
004439441035	11.2		3.8
004441443035	10.6	21.3	4.1
004441443535	10.6		4.1
004443445535	35.2	70.5	11.2
004443445035	35.2		11.2
004445447535	35.2	70.5	11.2
004445447035	35.2		11.2
002545547	1.7		
002546548	1.7		
002045047	1.7		
002046048	1.7		
002427527	5.70		
004427527027	900.	2300.	120.
004427527435	1400.		120.
002427528	5.70		
004427528027	900.	2300.	120.
004427528435	1400.		120.
002427027	5.70		
004427027527	900.	2300.	120.
004427027435	1400.		120.
002427028	5.70		
004427028527	900.	2300.	120.
004427028435	1400.		120.
002445545	3.14		
004445545045	6.3	54.3	14.4
004445545435	48.0		14.4
002445546	3.14		
004445546045	6.3	54.3	14.4
004445546435	48.0		14.4
002445045	3.14		
004445045545	6.3	54.3	14.4
004445045435	48.0		14.4
002445046	3.14		
004445046545	6.3	54.3	14.4
004445046435	48.0		14.4
004447449547	.01	347.0	.01
002527027	2.85		
002527028	2.85		
002528027	2.85		
002528028	2.85		
002546046	1.57		
002546045	1.57		
002545045	1.57		
002545046	1.57		
0101 .0063	.0063	.0736	
[illegible]	[illegible]	[illegible]	
0109 .0037	.0037	.0041	
0111 .0027	.0027	.0030	
0115 .0009	.0009	.0018	
0121 .0027	.0027	.009	
0125 .006	.006	.0123	
0128 .0058	.0058	.0106	
0129 .0122	.0122	.0122	
0131 .0069	.0069	.0112	

```
0132 .0035      .0035      .0157
0135 .0095      .0095      .0091
0139 .0142      .0142      .1256
0145 .0196      .0196      .1954
0151 .0031      .0031      .0237
0245 .0003      .0003      .0003
0246 .0003      .0003      .0003
0345 .0003      .0003      .0003
0346 .0003      .0003      .0003
0401 .0014      .0014      .0021
0405 .0043      .0043      .0035
0409 .0044      .0044      .0036
0412 .0026      .0026      .0057
0415 .0023      .0023      .0041
0419 .0087      .0087      .0109
0421 .0045      .0045      .0155
0423 .0049      .0049      .0210
0427 .0403      .0403      .0557
0431 .0092      .0092      .0822
0435 .0046      .0046      .0404
0439 .0033      .0033      .0193
0443 .0077      .0077      .0542
0445 .0085      .0085      .0632
0447 .0042      .0042      .0315
0527 .008       .008       .008
0528 .008       .008       .008
0027 .008       .008       .008
0028 .008       .008       .008
0545 .0005      .0005
0546 .0005      .0005
0045 .0005      .0005
0046 .0005      .0005                                                  9
99999
000138705.16        1.            8705.16
000551.             2.            3.            4.            5.
001056.             7.            8.            9.            10.
0015511.            12.           13.           14.           15.
0020516.            17.           18.           19.           20.
00355.03            .03           .03           .03           .03
00405.03            .03           .03           .03           .03
00455.03            .03           .03           .03           .03
00505.03            .03           .03           .03           .03
88888
00523-29.           -200.         -127.
01121121.
0114148.
88888
005210.
88888
000149838.          1.            9838.         1.
006511.
88888
00523-58.           -400.         -254.
01121242.
0114196.
88888
005210.
88888
0001420045.         1.            20045.        1.
88888
00523-120.          -832.         -529.
01121504.
01141201.
88888
005210.
88888
0001422655.         1.            22655.        1.
88888
00523-120.          -832.         -529.
```

```
01121504.
01141201.
88888
005210.
88888
0001419676.       1.          19676.       1.
88888
00523-87.         -600.       -382.
01123364.         0.          145.
88888
005210.
88888
0001429514.       1.          29514.       1.
88888
00523-80.         -555.       -353.
01123336.         0.          134.
88888
005210.
88888
000412.
88888
99999
```

PROGRAM DESCRIPTION

NASTRAN employs a lumped element approach, wherein the distributed physical properties of a structure are represented by a model composed of a finite number of idealized elements that are interconnected at a finite number of grid points. Loads are applied at these grid points, and all input and output data pertain to the idealized structural model. The steps in the definition and loading of a structural model are summarized in Figure 41.

The grid point definition forms the basic framework for the structural model. All other parts of the structural model are referenced either directly or indirectly to the grid points. A geometric grid point is a point in three-dimensional space at which three components of translation and three components of rotation are defined. The coordinates of each grid point are specified by the user.

Various kinds of constraints can be applied to the grid points. Constraints are used to specify boundary conditions, including enforced displacements of grid points, or to specify a linear relationship among selected degrees of freedom. Omitted points are used as a tool in matrix partitioning and for reducing the number of degrees of freedom used in dynamic analysis.

The elemental structural element is a convenient means for specifying many of the properties of the structure, including material properties, mass distribution, and some types of applied loads. Structural elements are defined on connection cards by referencing grid points. Generally, connection cards refer to property cards on which the cross-sectional properties of the element are specified. The property cards in turn refer to material cards which define the material properties.

The extensive computer-generated plotting capability available in NASTRAN has been used to plot the undeformed structural model and has provided for visual inspection of the model to check-out and debug the input data. Plots of the deformed structure have also been obtained to provide mode shape definition.

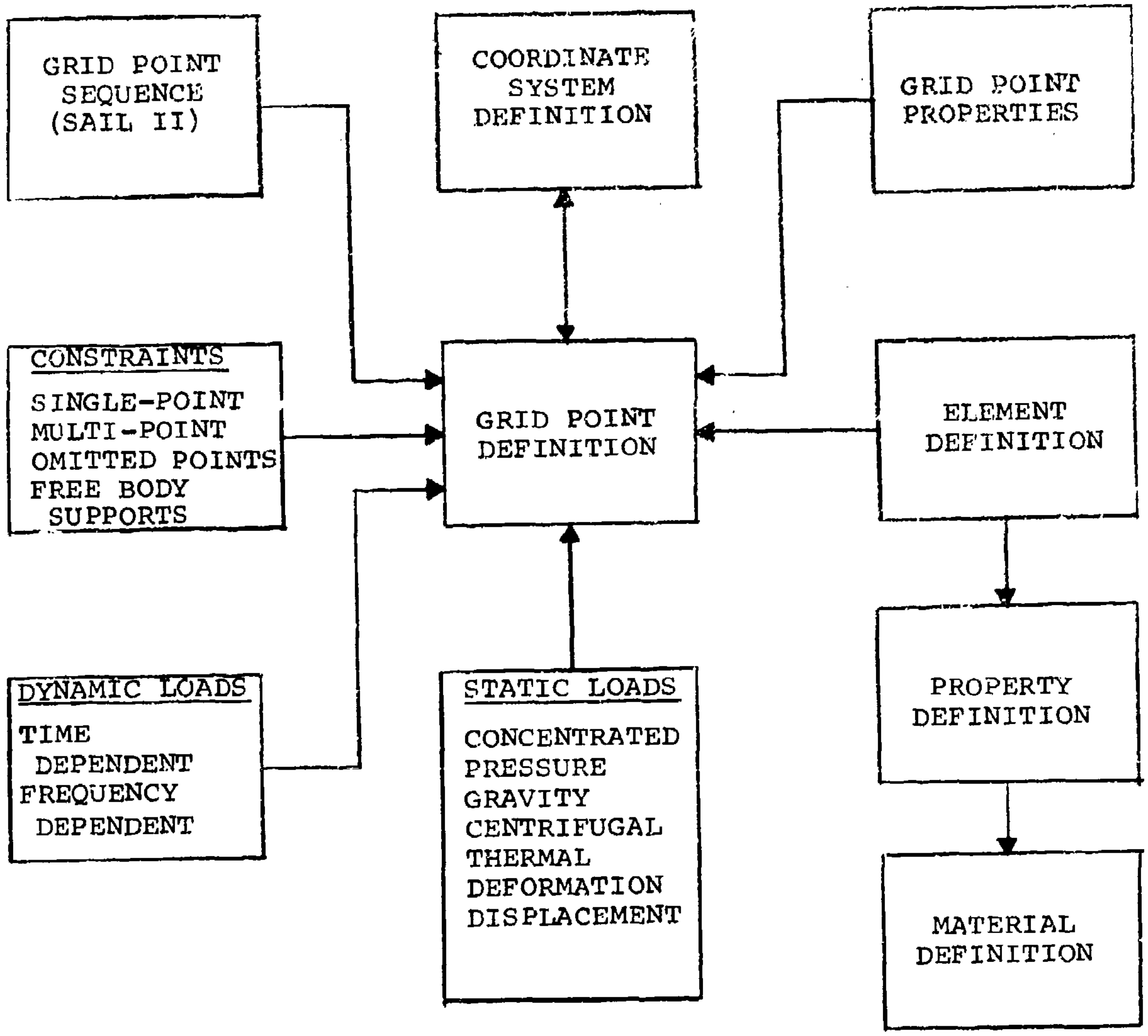


Figure 41. NASTRAN Structural Model Procedure.

PROGRAM UTILIZATION - CHECKOUT AND CORRELATION WITH SIREN TEST DATA

A cylinder has been modeled by Boeing Vertol using NASTRAN with the intent of determining the feasibility of using NASTRAN to dynamically analyze cylindrical shells. Since a survey of the literature indicated the lack of any other analytical model to correlate with the NASTRAN results, siren tests were conducted on an actual cylinder. Correlation was obtained after study through the use of a particular detailed analytical model of the cylinder, and it was concluded that NASTRAN was feasible to use. The results of the uniform cylinder test are summarized below.

A steel cylinder 6 inches high, 5.75 inches in diameter, and with a wall thickness of 1/8 inch was excited with the Boeing Vertol siren and its natural frequencies were measured. For comparison and correlation, a NASTRAN model was made of the test cylinder using three different mesh sizes. The model was comprised of homogeneous quadrilateral plate elements and was assumed to be a free-free body. A plane of symmetry was used to reduce the number of grid points to reduce computer run time. Grid points off the plane of symmetry were allowed inertial properties only in radial, azimuthal, and vertical directions. Rotations were omitted. The spacing between grid points was varied as 45, 22.5, and 10 degrees, and the model was eight plates high for each of the three spacings (Figure 42). The computer execution time for the various configurations is indicated by Figure 43.

The natural frequencies predicted by NASTRAN (Rigid Format 3) were compared to the experimentally measured frequencies. Good correlation was not obtained in the frequency range up to 3000 Hz until the grid spacing was reduced to 10 degrees. Table 7 and Figure 44 summarize the predicted and measured frequencies, and the first six mode shapes predicted by NASTRAN are shown in Figure 45.

An analytical solution for a uniform disc, which served as a second test case, is also available in Timoshenko. This was also modeled by Boeing Vertol using NASTRAN. The second diametral mode is shown in Figure 46. The reason for this correlation study was to justify the use of NASTRAN as a design tool for the dynamic analysis of turbine discs such as those used in the T55-L-11A gas turbine engine. The results of the Timoshenko analysis of the uniform disc compared well with the NASTRAN prediction. Figure 47 shows correlation of NASTRAN versus test for the third stage bolted turbine disc.

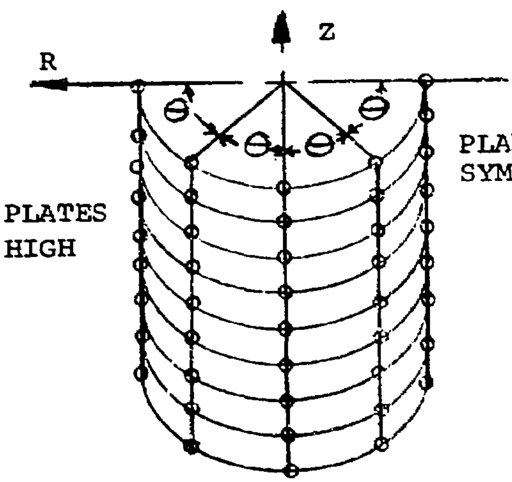


Figure 42. NASTRAN Model of Cylinder.

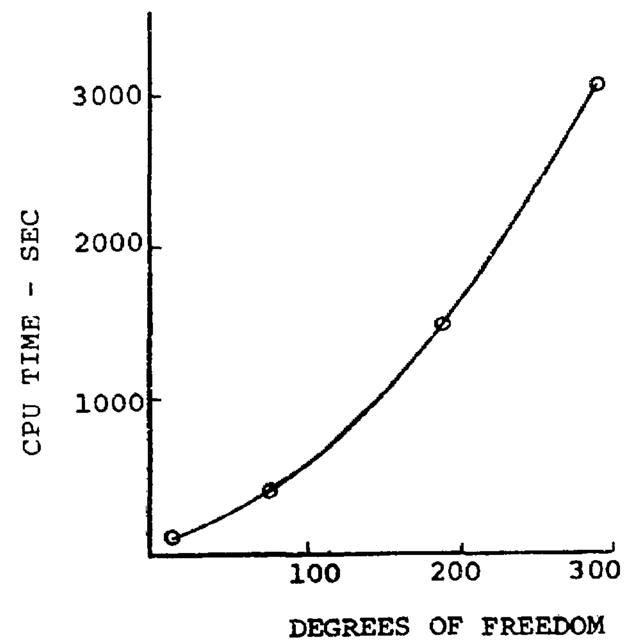


Figure 43. NASTRAN CPU Time.

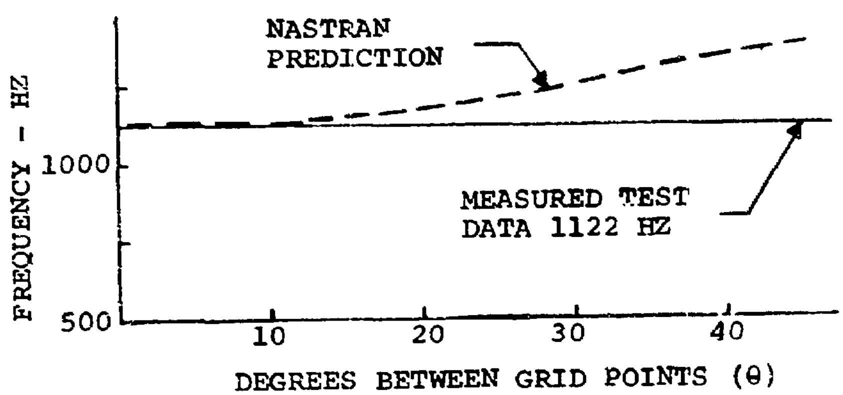


Figure 44. Convergence Plot.

TABLE 7. COMPARISON OF NASTRAN PREDICTED AND MEASURED FREQUENCIES.

	DEGREES BETWEEN GRID POINTS θ			TEST DATA	ERROR BETWEEN TEST AND θ = 10
	45°	22.5°	10°		
FREQUENCY - HZ	249	252	197		
	468	418	398		
	577	579	545		
	1393	1207	1128	1122	0.5%
	1463	1386	1320	1315	0.4%
	2147	2373	2160	2144	0.7%
	2165	2554	2382	2364	0.7%
	5930	3947	3495	3456	1.0%
		4100	3735	3688	1.2%
		4669	4291	4191	2.4%
			4781	4712	1.4%
			4979	4874	2.1%

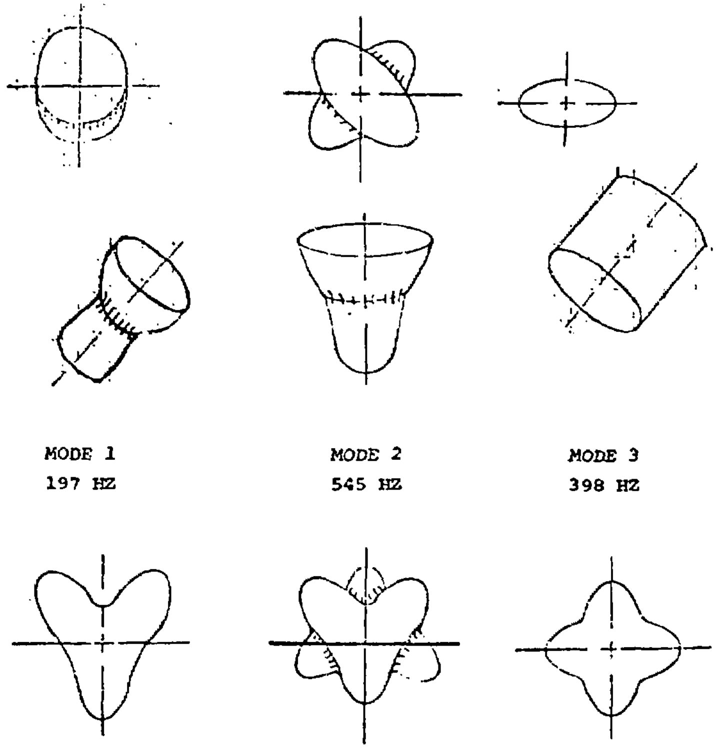
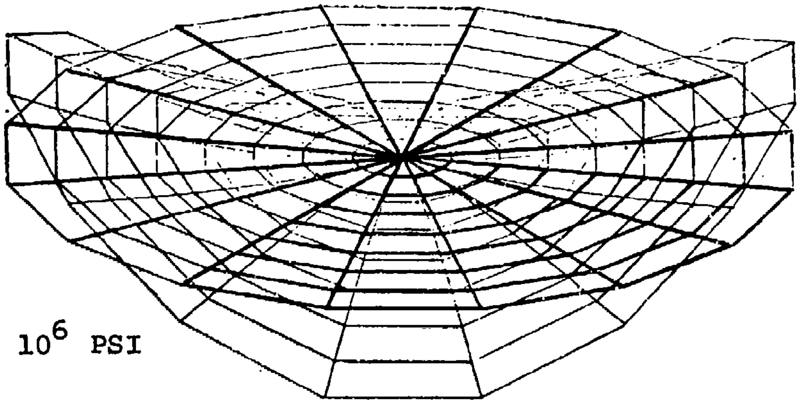


Figure 45. Predicted Mode Shapes for Cylinder.



$\rho = .286$

$\mu = .3$

$E = 30 \times 10^6$ PSI

$t = .25''$

Figure 46. NASTRAN Plot of Disc (2-D Mode).

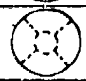
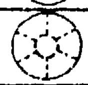
MODEL NO.			MODEL #1	MODEL #2	MEASURED DATA (ENGINE TEST)
BOUNDARY CONDITIONS			ATTACH TO TWO SHAFTS	FREE-FREE	
MODEL NO.	MODEL NAME	MODE SHAPE	CPS	CPS	CPS
1	FIRST DIAMETRAL		291	-	315
2	FIRST CIRCUM-FERENTIAL		361	776	
3	SECOND DIAMETRAL		415	362	400
4	THIRD DIAMETRAL		907	895	

Figure 47. T55-L-11A 3rd Stage Bolted Turbine Disk NASTRAN Natural Frequencies and Mode Shapes.

STRUCTURAL ANALYSES INPUT LANGUAGE PREPROCESSOR PROGRAM (SAIL II)

The contractor has developed pre- and post-processor computer programs compatible with NASTRAN to improve its utility. One of these is a sophisticated finite element input capability for use with NASTRAN entitled SAIL II (Structural Analyses Input Language). This preprocessor allows the user to take advantage of any pattern which occurs in the data by making available straightforward techniques for describing algorithms to generate blocks of data. Grid points and element connections may be generated. This program, although proprietary to the contractor, has been utilized and is available for purchase by industry. An alternative for nodal generation and/or connectivity would be user generated WATFOR computer programs which would punch out the NASTRAN input bulk data cards. Although the contractor has chosen to use SAIL II and feels that this program is more versatile, other users of NASTRAN can conduct the same work by the alternate method. The Boeing Vertol SAIL II computer program is compatible for use with NASTRAN Level 16.

For this contract, NASTRAN Level 15.5 was specifically used.

HELICOPTER TRANSMISSION HOUSING ANALYSIS (INCLUDING COMPOSITES)

In order to determine the dynamic or stress characteristics of the CH-47C forward transmission housing and to evaluate the use of various composite or isotropic materials, a finite element model of the housing, ring gear, and upper cover was constructed and analyzed. The modeling and analysis consisted of the following items:

1. Establishment of the grid point network to define the housing structural model.

2. Definition of the structural element connections and section properties of transmission housing. The housing was assumed in parts to be axisymmetric, and automatic node generation and structural connectivity computer programs were employed to generate the full transmission housing (e.g., SAIL II).

3. Definition of material properties of anisotropic or isotropic transmission housing by computing equivalent orthotropic material properties using an existing NASTRAN preprocessor (S-71, Point Stress Laminate Analysis) (Reference 20). This same computer program (S-71) may be used after the NASTRAN analysis as a post-processor to obtain interlaminar and laminar stresses in transmission

20. Reed, D.L., POINT STRESS LAMINATE ANALYSIS, Document FZM-5494, Prepared for Advanced Composite Division, Air Force Materials Laboratory, WPAFB, Ohio, April 1970.

housing areas of highest strain. Figure 48 illustrates the lamina or layer coordinate system (1-2) which is transformed by S-71 to the laminate (X-Y) axis system. In Figure 48, the resultant stresses and moments of the laminate are shown. These represent a system which is statistically equivalent to the stress system that is acting on the laminate. The notation used in computer program S-71 for a particular lamina within a laminate is also shown in Figure 48. The program will accommodate up to 400 layers. The equivalent material properties of the lamina may be calculated for the laminate. These would be used as input for the NASTRAN analysis. Also, a point stress analysis can be performed and thermal loads may be calculated. Lamina stresses and interlaminar shears can also be calculated.

4. Establish appropriate constraints for the 6 degrees of freedom at each grid point.

5. Optimize grid point sequencing and matrix bandwidth for efficient computer execution (BANDIT).

6. Plot and debug structural model.

7. Define and apply dynamic loads (bearing reaction) to model.

8. Run dynamic analysis (NASTRAN Rigid Formats 3 and 11) to predict dynamic deflections and loads.

Optimization techniques using a strain energy density approach have been developed to identify those areas of the housing which are most sensitive dynamically. The analysis as it is being applied to the CH-47C rotor transmission is outlined below.

The Boeing Vertol CH-47 forward rotor transmission housing is composed of three major sections: upper cover, ring gear, and case (Figure 49). The upper cover provides lugs for mounting the transmission to the airframe and transmits the rotor system loads. The case contains and supports the main bevel gears. The ring gear, which connects the upper cover and case, contains the planetary gear system. This natural division of the housing was adhered to for ease of modeling.

The geometric grid points for the model were defined from design drawings and by cross-checking on an actual housing. CQUAD2 (Quadrilateral) and CTRIA2 (Triangular) homogeneous plate (membrane and bending) elements were used to connect the grid points and build the NASTRAN structural model. A Boeing Vertol preprocessor program (SAIL II - Structural Analyses Input Language) for the automatic generation of grid point coordinates and structural element connections was used.

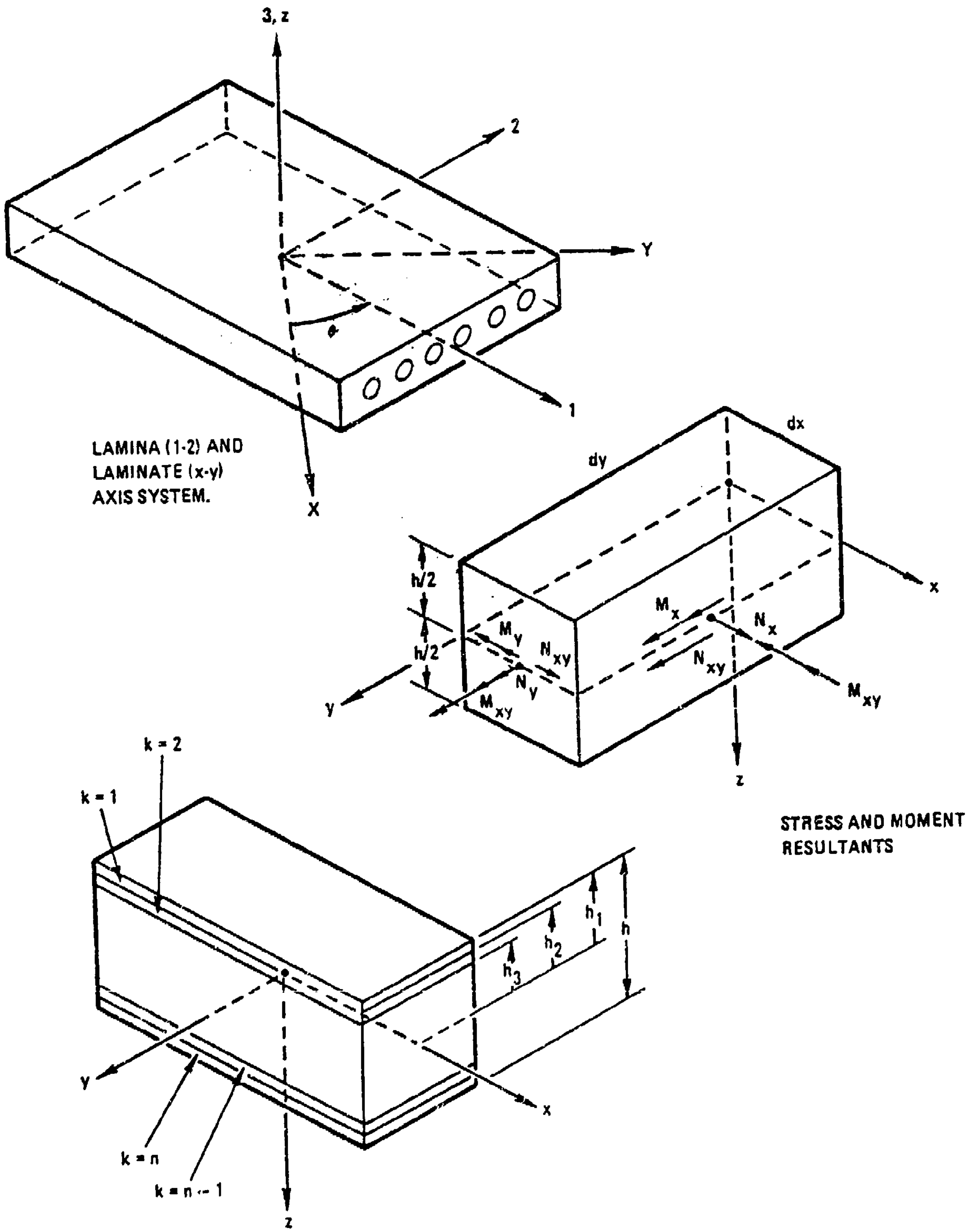


Figure 48. Lamina Notation.

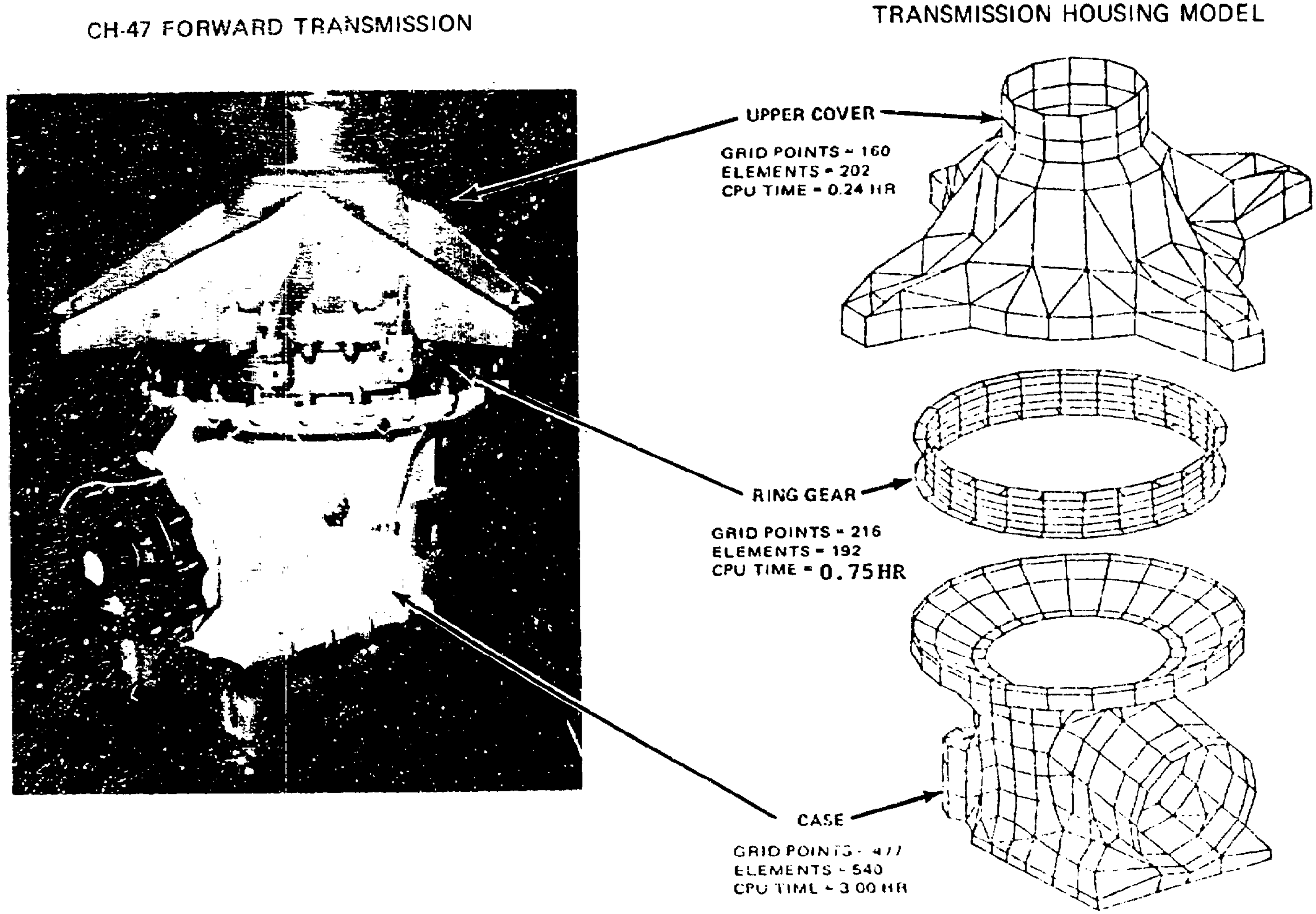


Figure 49. Boeing Vertol CH-47 Helicopter Forward Rotor Transmission Housing and NASTRAN Model.

This preprocessor allows the user to take advantage of any pattern which occurs in the data by providing straightforward techniques for describing algorithms to generate blocks of data. The extensive computer-generated plotting capability of NASTRAN was used to debug the structural model.

For ease of identification the housing was subdivided into several regions and the grid points in each region were labeled with a specific, but arbitrary, series of numbers. Although these grid point numbers act only as labels, they affect the bandwidth of the stiffness and mass matrices. In order to minimize the matrix bandwidth for most efficient running of NASTRAN, the BANDIT computer program (Reference 21) was used to automatically renumber and assign internal sequence numbers to the grid points. The output from BANDIT is a set of SEQGP cards which are then included in the NASTRAN bulk data deck and which relate the original external grid numbers of the internal numbers.

The model includes grid points representative of the structure where the shafts are supported by their bearings. These grid points are used to apply the dynamic excitations at the mesh frequencies to analytically excite the housing. Although each geometric grid point has six possible degrees of freedom (three translational and three rotational), the displacements normal to the outer surface of the housing are of most interest for noise evaluation since it is this out-of-plane motion which generates sound waves. To conveniently evaluate the motion normal to the housing surface, numerous local coordinate systems were defined and oriented such that the displacements and accelerations calculated at each grid point could be referred to a coordinate system having an axis normal to the housing surface (Figure 50). One degree of freedom, rotation about the normal to the surface, was constrained since the stiffness for this component is undefined for NASTRAN plate elements. The other two rotational degrees of freedom were omitted. All translational degrees of freedom were retained to accurately represent the motion of the actual housing. Because of the large model size, the Guyan reduction technique was used to reduce the size of the analysis set. The Givens method of eigenvalue extraction was used and the model parameters are summarized in Figure 51. The NASTRAN weight generator feature was also used to calculate the model weights, which are also summarized in Figure 51.

The contractor has developed confidence that the modeling procedure provides an accurate representation of the actual transmission housing for the following reasons:

---

21. Everstine, G., BANDIT - A COMPUTER PROGRAM TO RENUMBER NASTRAN GRID POINTS FOR REDUCED BANDWIDTH, Naval Ship Research and Development Center Technical Note AML-6-70, February 1970.

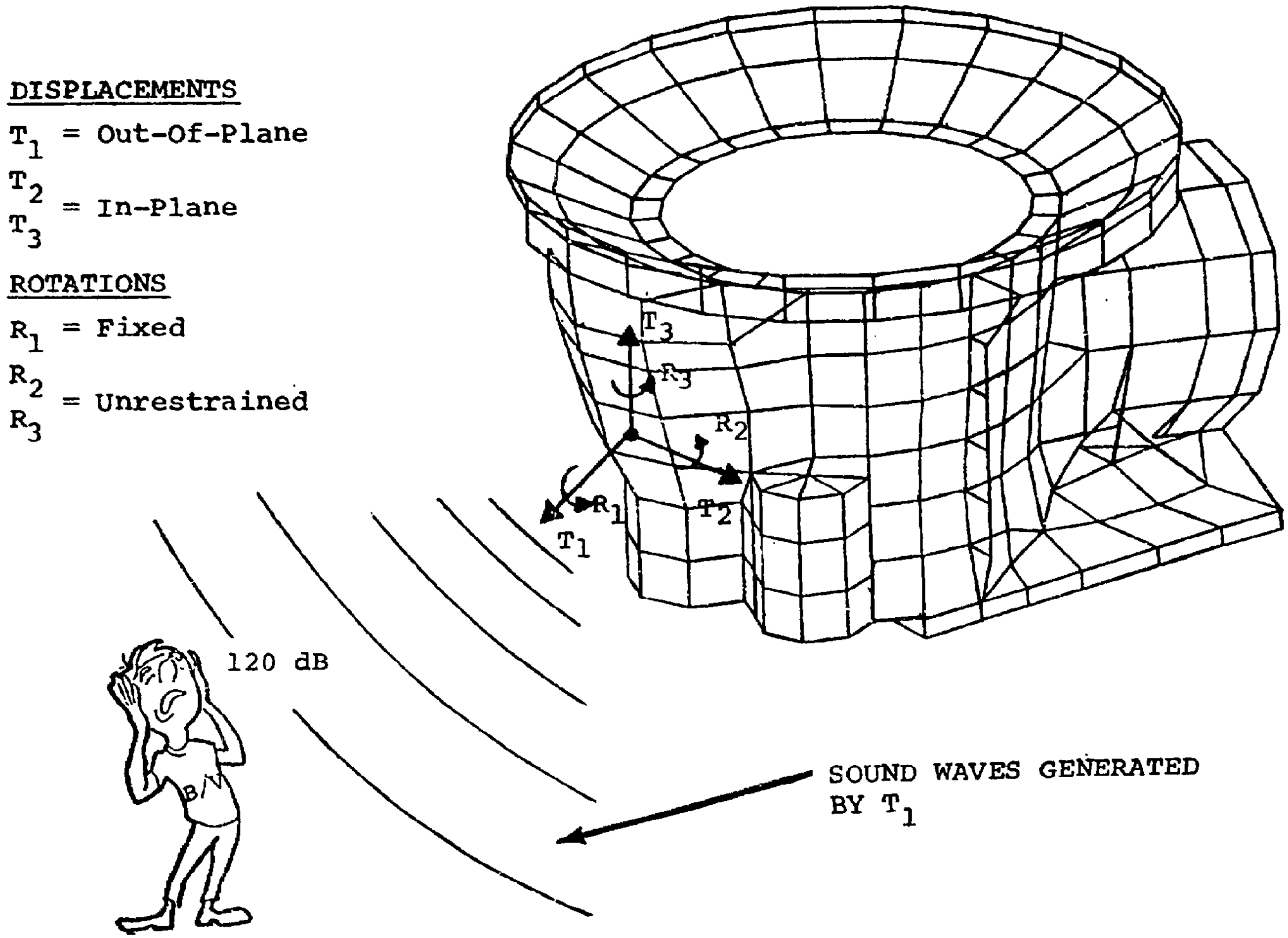


Figure 50. Transmission Noise Generated by Out-Of-Plane Displacements of Housing.

MODEL PARAMETERS

	NUMBER GRID POINTS	NUMBER ELEMENTS	NUMBER DEGREES OF FREEDOM				BANDWIDTH			CPU TIME (HOURS)*
			TOTAL	SPC	OMIT	RETAINED	FULL	REDUCED	ACTIVE COLUMNS	
Upper Cover	160	202	960	184	614	162	34	162	0	0.24
Ring Gear	216	192	1296	216	828	252	-	252	0	0.75
Case	477	540	2862	529	2024	309	61	309	0	3.00
TOTAL	853	934								

*RIGID FORMAT 3 ON IBM 370

COMPARISON OF CALCULATED AND ACTUAL WEIGHTS*

	MODEL	HARDWARE	DIFFERENCE
Sump	3.7 kg (8.2 lb)	5.5 kg (12.2 lb)	**
Case	25.1 kg (55.4 lb)	24.6 kg (54.2 lb)	+ 2.2%
Ring Gear	34.9 kg (77.0 lb)	34.9 kg (77.0 lb)	0% (Lumped Masses for Teeth)
Upper Cover	62.8 kg (138.5 lb)	64.1 kg (141.4 lb)	- 2.0%

*(Case weight based on AZ91C cast magnesium alloy, density .065 lb/in^3; upper cover weight based on 2014-T6 forged aluminum, density .101 lb/in^3 , both per MIL-HDBK-5B 1 September 71.)

**Model excludes internal passageways.

Figure 51. Summary of CH-47 Forward Transmission Housing NASTRAN Model.

- Use of a widely accepted and thoroughly validated computer program (NASTRAN).
- Extensive computer-generated plotting capability used to debug model.
- Cross-checking of model, design drawings, and hardware.
- Good correlation of model and hardware weights.
- Further validation is ongoing in the form of correlation of dynamic response of model with test data.

Each natural mode of a structure contributes to vibration in proportion to its amplification factor. Consequently, since each mode whose frequency is in the vicinity of a forcing frequency will be a major contributor to the overall dynamic response, it is desirable to alter the housing natural frequencies so that none fall close to an exciting frequency.

Strain energy techniques for structural optimization have evolved in recent years. For applications such as helicopters where weight is critical, it is more appropriate to evaluate the strain density (strain energy/volume) distribution within a structure which provides guidance for vibration reduction by identifying the structural elements participating in the modes. To optimize a housing for minimum vibration/noise, the NASTRAN normal modes analysis is used to obtain a dynamic solution; by employing the ALTER feature of NASTRAN, a checkpoint tape containing the stresses for each element is generated. The natural frequencies calculated are compared with the gear mesh exciting frequencies to identify each mode shape whose natural frequency is close to an exciting frequency and which it is desirable to shift. A post-processor program (S-83) has been developed which uses the data stored on the checkpoint tape to calculate the strain density of NASTRAN plate elements and tabulate the elements in order of descending strain density. The structural elements with the highest strain density are the best candidates for effective modification of the natural frequency, since a minimal weight change will yield a maximum shift in natural frequency. By locally altering the housing wall to change the mass and stiffness in these areas of high strain density, the natural frequency may be shifted away from an exciting frequency. Thus, the possibility of resonance is eliminated and the vibration and radiated noise are reduced. This strain density distribution concept will also be utilized statically to identify structural load paths and evaluate the efficiency of the housing structural design (stiffness/weight). By controlling this energy distribution, stress, vibration/noise, and weight may be reduced.

Figures 52, 53, and 54, respectively, show a NASTRAN plot of the housing (left and aft views), some typical natural frequencies in the vicinity of meshing frequencies, and a NASTRAN plot of a typical mode shape.

NASTRAN, LEVEL 16

With the advent of NASTRAN, Level 16, with its capability to calculate strain energy, it may be used in place of D-82, S-68, and S-83 for all calculations (shafts and housing). This can be done as follows:

1. Conduct a dynamic analysis (Rigid Format 3) for shafts and housing. Punch out eigenvectors.

2. Reformat the punched output to SPC type input. Execute Rigid Format 1 to obtain strain energy for structural elements.

Two post-processor computer programs were developed to expedite and expand upon the above-mentioned procedures. A listing of a computer program for use with NASTRAN, Level 16 in order to take punched strain energy output from Rigid Format 1 and sort and analyze is shown in Appendix F. Appendix G contains the listing of a second computer program to be used with NASTRAN, Level 16, to reformat Rigid Format 3 to punched eigenvector output for execution with Rigid Format 1 as SPC cards for strain energy calculations. A sample strain energy output which was sorted and analyzed using the above programs is shown in Figure 55.

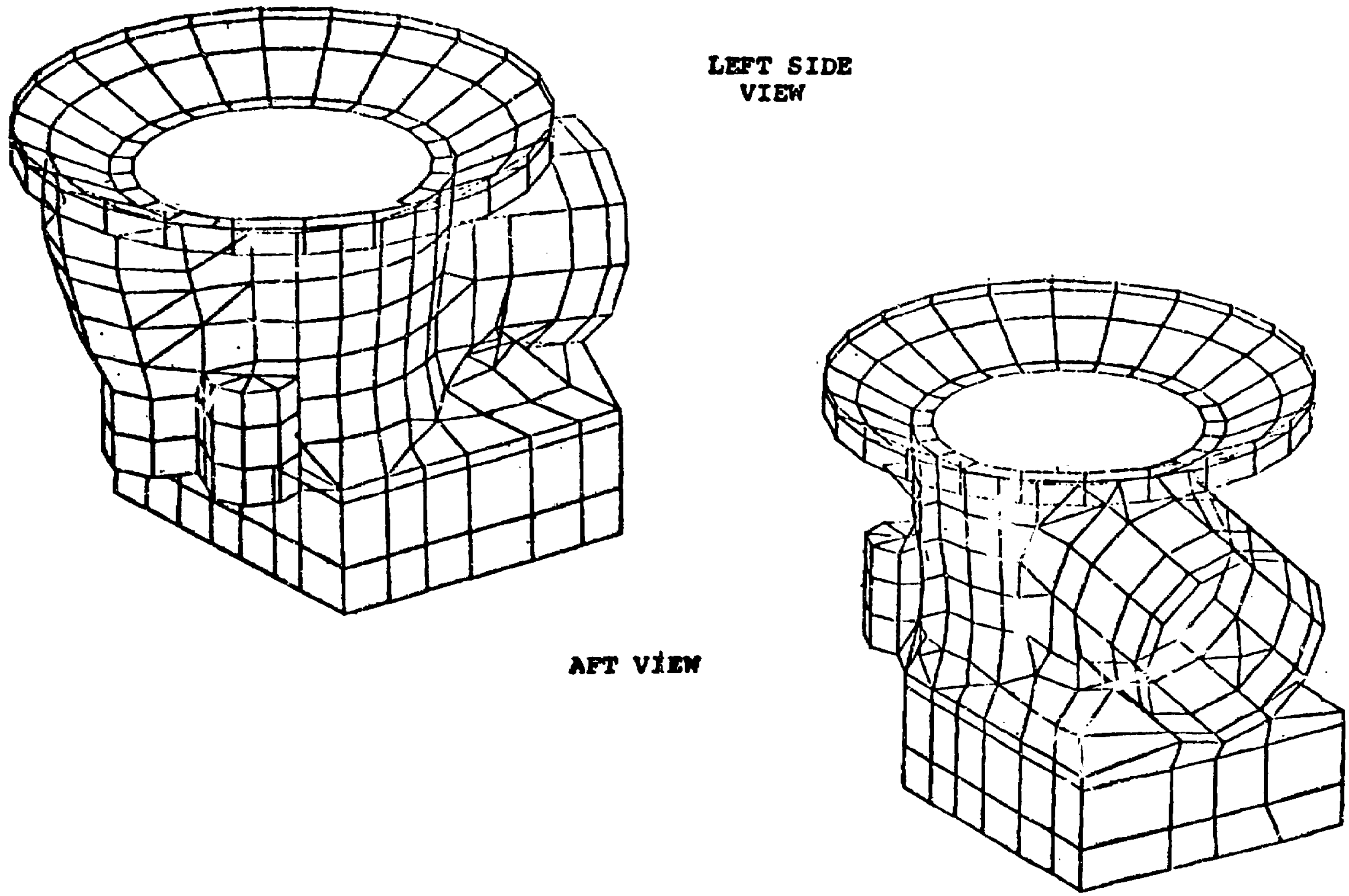


Figure 52. Computer-Generated Plots of NASTRAN Model; CH-47C Forward Rotor Transmission Case with Sump.

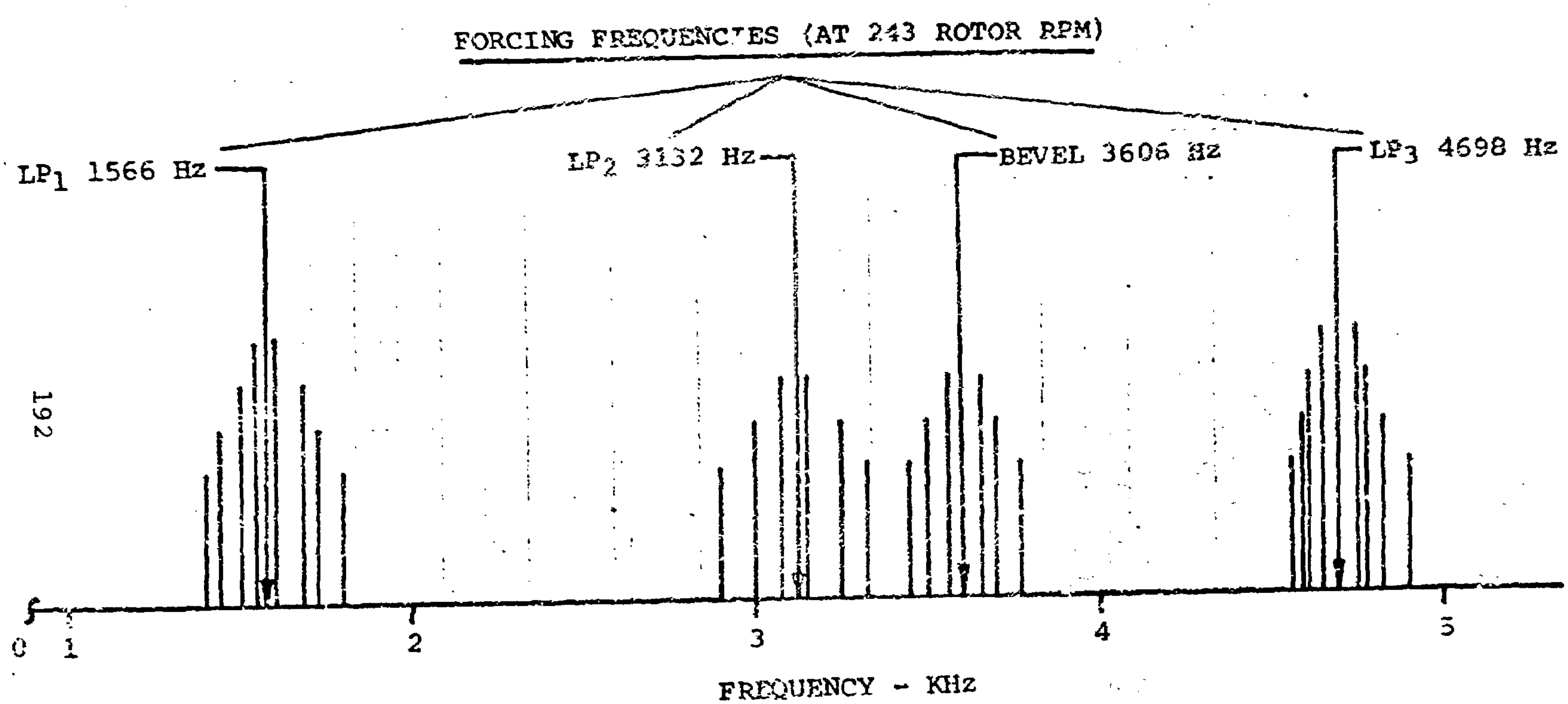


Figure 53. Spectrum of Forcing Frequencies Versus NASTRAN Predicted Natural Frequencies for CH-47C Forward Transmission Case.

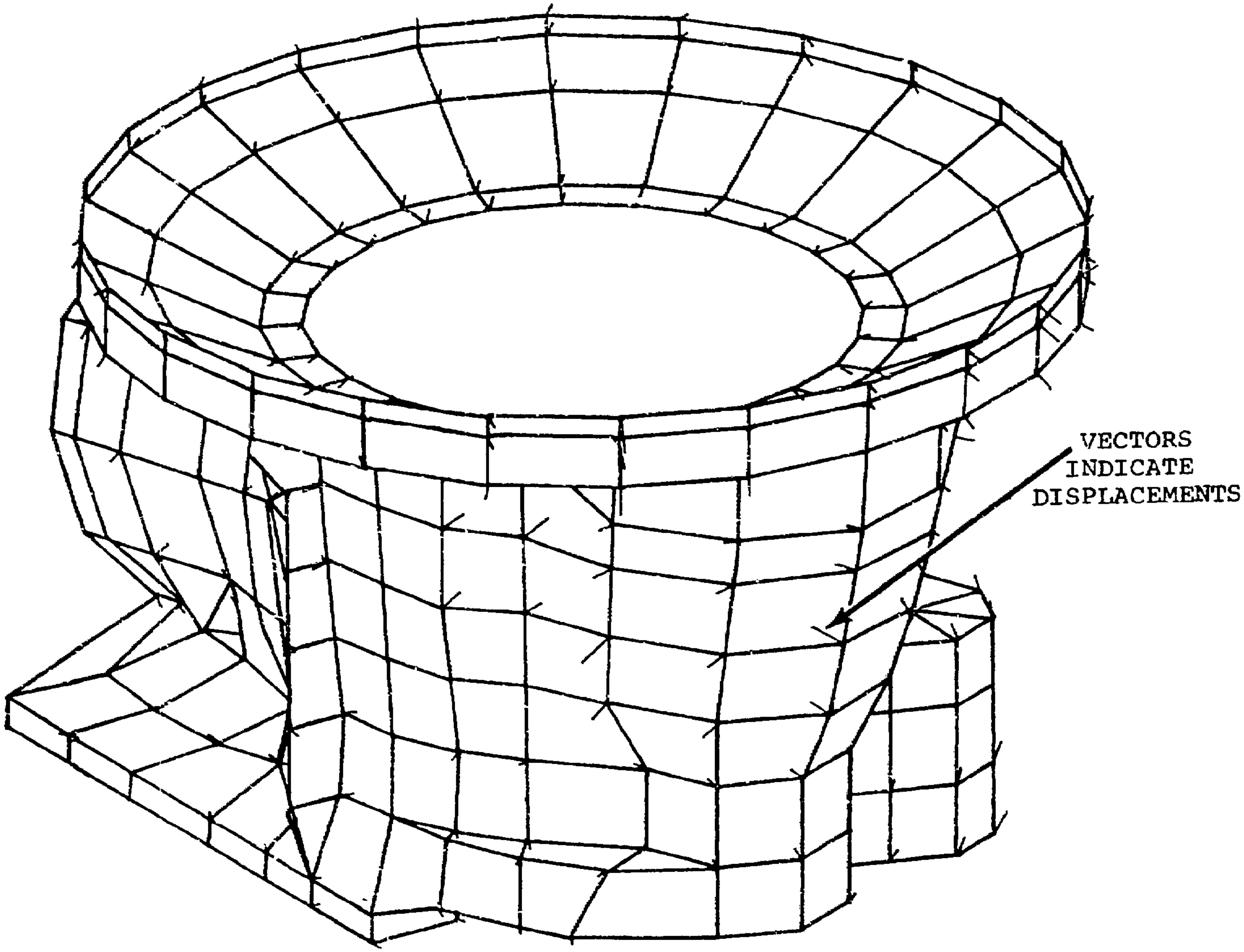


Figure 54. NASTRAN Plot of Deformed Housing,
Mode #46, Frequency 3141 Hz.

TITLE F FULLY STRESSED DESIGN	106
SUBTITLE	108
LABEL F	111
ELEMENT STRAIN ENERGIES	112
REAL OUTPUT	113
SUBCASE 1	114

INDEX NO.	ELEMENT NO.	STRAIN	PERCENT	ACCUMULATED PERCENT
1	6011	426.75036	5.45636	5.45636
2	77	275.14745	3.51815	8.97451
3	6016	271.44727	3.47084	12.44535
4	6017	267.60219	3.42165	15.86700
5	49	188.74873	2.41482	18.27582
6	66	184.31792	2.35675	20.63257
7	105	178.79659	2.28093	22.91348
8	6001	173.14851	2.22296	25.13637
9	86	159.48029	2.03918	27.17554
10	87	159.01039	2.03317	29.20871
11	88	158.27710	2.02380	31.23250
12	[illegible]	154.02103	1.96936	33.20187
13	63	153.02069	1.95821	35.17007
14	71	146.79053	1.87577	37.04584
15	6022	144.94710	1.85335	38.89919
16	4016	143.92229	1.79293	40.69212
17	5011	133.49756	1.70596	42.39808
18	6018	118.62336	1.51676	43.91585
19	5016	118.09569	1.50887	45.42470
20	6006	112.71973	1.44128	46.86597
21	66	109.78609	1.40377	48.26975
22	68	105.13770	1.34505	49.61478
23	63	96.91559	1.23920	50.85397
24	85	95.87067	1.22584	52.07981
25	5006	94.33553	1.20621	53.28601
26	4016	92.72202	1.18558	54.47159
27	101	91.87479	1.17475	55.64633
28	97	81.17059	1.03798	56.68431
29	4017	75.87186	0.97013	57.65443
30	5011	72.61134	0.92854	58.58298
31	51	71.78740	0.91790	59.50088
32	77	71.16741	0.90998	60.41083
33	301	65.82515	0.84167	61.25249
34	84	64.72321	0.82758	62.08006
35	52	60.94674	0.77928	62.85924
36	21	60.73176	0.77526	63.63460
37	304	59.64013	0.76003	64.39462
38	70	58.86732	0.75271	65.14732
39	307	58.65146	0.74994	65.89720
40	74	56.75658	0.72550	66.62276
41	67	52.94136	0.67694	67.29968
42	3	52.92777	0.67676	67.97643
43	4022	51.15437	0.65408	68.63049
44	70	49.67717	0.63798	69.26846
45	4007	49.23338	0.62950	69.89524
46	61	49.05011	0.62601	70.52325
47	313	48.41152	0.61901	71.14225
48	4007	48.06731	0.61844	71.77280
49	317	48.07007	0.61475	72.38744

Figure 55. Sample Strain Energy Output (Sorted and Analyzed).

50	31[illegible]	47.66133	0.60175	72.96918

TITLE = FULLY STRESSED DESIGN 110
SUBTITLE= 110
LABEL = 111
ELEMENT STRAIN ENERGIES 112
REAL OUTPUT 113
SUBCASE ID 4 114

INDEX NO.	ELEMENT NO.	STRAIN	PERCENT	ACCUMULATED PERCENT
51	67	45.52327	0.58206	73.57126
52	31[illegible]	45.[illegible]9706	0.57548	74.14673
53	6012	44.88529	0.57392	74.72064
54	41	44.44524	0.56830	75.28893
55	56	42.43554	0.54260	75.83151
56	66	41.63471	0.53238	76.36389
57	1[illegible]	41.[illegible]97[illegible]	0.52808	76.89197
58	76	[illegible]0.[illegible]679[illegible]	0.51130	77.40326
59	96	39.[illegible]51[illegible]	0.50700	77.91025
60	75	38.64684	0.49414	78.40439
61	5018	34.54166	0.48281	78.88719
62	4018	37.41527	0.47841	79.36558
63	18	36.11526	0.46165	79.83723
64	26	35.48252	0.45370	80.29092
65	92	35.13773	0.44990	80.73982
66	64	35.05736	0.44821	81.19863
67	5012	31.28590	.39957	81.58860
68	16	30.91196	0.39524	81.98323
69	59	30.24854	0.38677	82.37 [illegible]0
70	321	30.21233	0.38631	82.7563[illegible]
71	315	29.46791	0.37335	83.13564
72	17	29.61179	0.37863	83.51425
73	5017	29.15656	0.37273	83.88691
74	6017	28.49257	0.37355	84.25053
75	318	28.35919	0.36236	84.61666
76	78	28.01282	0.35918	84.97765
77	108	27.49232	0.35038	85.32742
78	81	27.11313	0.34668	85.67415
79	44	27.03011	0.34562	86.01971
80	91	25.92763	0.33615	86.34985
81	342	25.50087	0.32626	86.67590
82	78	25.48792	0.32590	87.00180
83	327	23.57674	0.30146	87.30325
84	1	22.57373	0.28871	87.59196
85	5013	21.58124	0.27594	87.86791
86	63	20.[illegible]14	0.26832	88.13622
87	6015	20.54959	0.26275	88.39896
88	6021	20.38536	0.26066	88.65961
89	805	20.21687	0.25850	88.91811
90	73	19.99973	0.25573	89.17385
91	6001	19.84424	0.25374	89.42759
92	6021	19.49747	0.24929	89.67687
93	312	19.29311	0.24669	89.92356
94	15	18.27013	0.23361	90.15717
95	72	17.85712	0.22833	90.38544
96	16	17.73912	0.22682	90.61231
97	6012	17.22349	0.22112	90.83463
98	5023	17.12435	0.21895	91.05357
99	47	16.89757	0.21606	91.26962

Figure 55. Continued.

100	322	16.75165	0.21419	91.48361
101	5005	16.74265	0.21417	91.69757

TITLE = FULLY STRESSED DESIGN 106
SUBTITLE= 108
LABEL = 111
ELEMENT STRAIN ENERGIES 112
REAL OUTPUT 113
SUBCASE ID = 1 114

INDEX NO.	ELEMENT NO.	STRAIN	PERCENT	ACCUMULATED PERCENT
102	701	16.46004	0.21046	91.90843
103	29	16.04616	0.20517	92.11360
104	4012	15.73586	0.20120	92.31479
105	42	15.60755	0.19951	92.51430
106	173	15.54655	0.19878	92.71307
107	29	15.50766	0.19828	92.91135
108	24	15.29833	0.19563	93.10696
109	25	14.59974	0.18669	93.29364
110	5007	13.82959	0.17671	93.47035
111	6011	13.05456	0.16692	93.63727
112	22	12.76445	0.16321	93.80048
113	12	12.19447	0.15592	93.95639
114	306	12.17999	0.15574	94.11212
115	6005	12.07572	0.15441	94.26653
116	65	12.05874	0.15419	94.42070
117	4007	11.63115	0.14872	94.56941
118	314	11.61799	0.14856	94.71791
119	5011	11.11485	0.14212	94.86002
120	21	11.10542	0.14200	95.00201
121	35	10.85921	0.13885	95.14085
122	6004	10.38404	0.13277	95.27362
123	58	10.30713	0.13179	95.40541
124	5006	9.74909	0.12466	95.53006
125	14	9.60355	0.12279	95.65285
126	13	9.52035	0.12173	95.77457
127	4011	9.34389	0.11947	95.89403
128	49	8.95543	0.11451	96.00853
129	5014	8.74493	0.11181	96.12033
130	43	8.73700	0.11171	96.23204
131	2	8.47608	0.10838	96.34041
132	6014	8.37788	0.10712	96.44753
133	707	8.10700	0.10366	96.55118
134	94	8.00674	0.10238	96.65355
135	4013	7.84901	0.10036	96.75391
136	107	7.64464	0.09781	96.85172
137	801	7.54196	0.09643	96.94814
138	4005	7.11464	0.09097	97.03909
139	55	6.87924	0.08796	97.12704
140	4	6.80863	0.08706	97.21410
141	5008	6.55646	0.08383	97.29793
142	106	6.52665	0.08345	97.38138
143	66	6.52399	0.08342	97.46478
144	5015	6.40151	0.08185	97.54663
145	3	6.36879	0.08143	97.62805
146	35	6.23876	0.07977	97.70781
147	54	6.21535	0.07947	97.78728
148	27	5.99641	0.07667	97.86394
149	95	5.70743	0.07298	97.93690

Figure 55 Continued.

150	9	5.65877	0.07236	98.00925
151	710	5.37497	0.06873	98.07797
152	93	4.98250	0.06371	98.14168

106

TITLE = FULLY STRESSED DESIGN 108
SUBTITLE= 111
LABEL = 112
ELEMENT STRAIN ENERGIES 113
REAL OUTPUT 114
SUBCASE ID = 0

INDEX NO.	ELEMENT NO.	STRAIN	PERCENT	ACCUMULATED PERCENT
153	57	4.96290	0.06346	98.20512
154	5	4.95967	0.06342	98.26854
155	5009	4.95411	0.06335	98.33188
156	39	4.88310	0.06244	98.39430
157	311	4.82462	0.06169	98.45598
158	104	4.81942	0.06162	98.51759
159	48	4.77685	0.06108	98.57866
160	704	4.74314	0.06065	98.63930
161	62	4.65905	0.05957	98.69887
162	[illegible]	4.61252	0.05898	98.75784
163	10	4.59195	0.05871	98.81654
164	36	4.56091	0.05832	98.87485
165	6022	4.51710	0.05776	98.93260
166	92	4.48036	0.05729	98.98988
167	305	4.33798	0.05547	99.04535
168	105	4.18970	0.05357	99.09891
169	6020	4.06903	0.05203	99.15092
170	27	3.62178	0.04631	99.19722
171	8	3.55622	0.04547	99.24269
172	11	3.39886	0.04346	99.28615
173	705	3.38133	0.04324	99.32938
174	7	3.31455	0.04238	99.37175
175	309	3.29408	0.04212	99.41386
176	4009	3.04433	0.03893	99.45279
177	706	2.92461	0.03740	99.49017
178	712	2.86452	0.03663	99.52679
179	4015	2.85441	0.03650	99.56328
180	45	2.76855	0.03540	99.59868
181	46	2.73507	0.03497	99.63364
182	28	2.70035	0.03453	99.66815
183	34	2.45320	0.03137	99.69951
184	33	2.32077	0.02967	99.72917
185	803	2.30650	0.02949	99.75865
186	6	2.26079	0.02891	99.78755
187	6021	2.22328	0.02843	99.81598
188	32	2.15386	0.02767	99.84364
189	31	1.81150	0.02316	99.86679
190	40	1.74836	0.02236	99.88914
191	702	1.61172	0.02061	99.90974
192	4022	1.53995	0.01969	99.92943
193	804	1.50761	0.01928	99.94870
194	802	1.45147	0.01856	99.96725
195	711	0.85302	0.01091	99.97815
196	703	0.52720	0.00674	99.98488
197	4014	0.47319	0.00605	99.99092
198	4120	0.46522	0.00595	99.99686
199	4021	0.26334	0.00337	100.00021

Figure 55. Continued.

STRAIN ENERGY ANALYSIS PROGRAMS (S-68 AND S-83)

PROGRAM DESCRIPTION

Engineering judgment has traditionally provided an approach to the optimization of a helicopter structure for vibration. This requires the engineer to have a prior comprehensive understanding of the analytical model, i.e., the eigenvalues and eigenvectors of the structure together with their influence on critical structural elements. Many cases of structural changes would be evaluated using the finite element dynamic solution, and the most favorable changes for the final design would be chosen from the study. This has proved to be a virtually impossible method, however, because the enormous number of structural elements and the need for side constraints make the number of possibilities for structural changes quite large. This approach is limited in that it only provides for studies of known structural change effects. For a new design, identification of the structural components that have a critical influence on vibration would be difficult. Also, no criteria exist to ensure that the final structural changes are optimum. The engineering judgment approach at best provides a minimal analytical tool for vibration design of new structures.

In recent years, a trend to dynamic optimization by strain energy techniques has evolved. This has mainly been for the optimal alteration of one undesirable natural frequency. To understand the technique, consider that in general each natural mode of the structure contributes to vibration in proportion to its amplification factor. Consequently, each mode whose frequency is in the vicinity of the forcing frequency would be a major contributor to the overall dynamic response. In the modal method, which operates principally on the amplification factor, the natural frequency immediately above the exciting frequency is usually increased. One could also reduce the natural frequency immediately below the exciting frequency if it is possible structurally.

Analysis has previously been developed verifying that a minimum weight structure with a specified natural frequency is one wherein the density differential is uniform throughout the structure when deformed in its natural mode (Reference 6). The density differential of a structural element is the difference between the strain energy per unit volume (strain density) and the kinetic energy per unit volume (kinetic density). In most cases, the strain density may be used as an approximation of the density differential, since the kinetic density is relatively small. Since the objective for aircraft is a maximum eigenvalue shift for a minimum

weight change, the strain energy density is used rather than the strain energy itself. It follows that a complex structure can most efficiently be redesigned dynamically by ensuring that the modal strain density is uniform throughout the structure. Hence, vibration optimization of a structure can be put on a more scientific basis if one considers its modal strain energy or density distribution.

A finite element analysis is first employed in the modal method to yield a dynamic solution. The eigenvectors (mode shapes) are obtained, then the modal strain energy distribution throughout the structure is found for any given mode shape whose natural frequency is to be modified. The strain energies for all structural elements are obtained and then tabulated from the highest to lowest. The structural elements in the highest strain would be the best candidates for modification of the natural frequency. This follows intuitively in the modal case from the almost invariency of the mode shapes and from Rayleigh's quotient. Theory also substantiates this in that it has been proven analytically that the rate of change of a natural frequency (eigenvalue) with respect to a structural parameter (thickness herein) of an element is equal to the strain energy of the structural element when the entire structure is deformed according to the mode shape. For example, in the case of increasing the lowest mode, the elements with the highest strain density, when deformed in this mode, would be the best candidates for modification to obtain a maximum eigenvalue shift for a minimum weight penalty (i.e., optimally).

Two computer programs have been utilized to evaluate the strain energy of the transmission components. Using the mode shapes calculated in D-82 for the shaft model, program S-68 has been used to identify the areas of high strain energy density for the shafting. A listing of this program is contained in Appendix H. S-68 computes the normalized strain energy for each structural element using the critical modes independently. Then the strain density of each element is determined as a function of its volume. These are then sorted from the highest to lowest strain density, and the elements with high strain density are then the optimal members to change for a minimum weight penalty in order to shift the natural frequency. This program was developed by Boeing Vertol under Army Research Office contract DAHC04-71-C-0048. A second strain energy program (S-83) compatible with NASTRAN (plate elements) has been developed and used to identify the listing of this program is contained in Appendix I. These areas are directly associated with undesirable vibrations and noise. Though variations exist in wall thickness and geometry, these regions can be modified to provide an improved design configuration. To optimize a housing for minimum vibration/noise, the NASTRAN normal modes analysis is used to obtain a

dynamic solution. By employing the ALTER feature of NASTRAN, specifically for Rigid Format 3,

```
ALTER 107

CHKPNT OES1, OEF1 $

ENDALTER
```

a checkpoint tape containing the stresses (data block OES1) and forces (data block OEF1) for each element is generated. These should come last in the case control. "ELFORCES" and "STRESSES" should also be calculated. The S-83 post-processor program reads the modal dynamic stresses from the checkpoint tape for each mode shape, calculates the strain density of each of the NASTRAN plate elements,

$$S.D. = \frac{S.E.}{V} = \frac{1}{2E} \left(\sigma^2_{max} + \sigma^2_{min} - 2\mu\,\sigma_{max}\,\sigma_{min}\right) \frac{1}{V}$$

where

S.D. is the strain density,

S.E. is the strain energy,

E is Young's modulus,

σ_{max}, σ_{min} are principal stresses,

μ is Poisson's ratio, and

V is the volume of the element,

and then tabulates the elements in order of descending strain density.

In the calculation of the plate stresses, NASTRAN calculates the stresses for the upper and lower surfaces. S-83 has options whereby the strain density calculation may be based on upper stresses, lower stresses, or membrane stresses, where

$$\sigma_u = \sigma_b + \sigma_a$$

$$\sigma_L = \sigma_b - \sigma_a$$

yields

$$\sigma_b = \frac{\sigma_u + \sigma_L}{2}$$

$$\sigma_a = \frac{\sigma_u - \sigma_L}{2}$$

and where the subscripts

u is upper,

L is lower,

b is bending, and

a is membrane.

The above separation of bending and membrane stress would yield another and more exacting strain density analysis. For the housing, the outer combined bending-membrane stresses were used.

In order to determine whether strain energy methods could be used for the vibration reduction of a large, complex structure, some evidence as to feasibility was needed. These methods have been successfully applied to the vibration reduction of the Boeing Vertol Model 347 helicopter. Impedance methodology (applied to the design of absorbers) and detuning of the 347 fuselage (by structural modification) have resulted in an exceptionally low final vibration level throughout the entire flight envelope. The structure to be modified for detuning was indicated by the modal strain energy technique. Hence, the feasibility of strain energy methods was established as a practical means for dynamic design for large complex structures (Reference 5). Figures 56 and 57 show this (analytically) for the CH-47C forward transmission ring gear.

This strain density distribution concept can also be utilized statically to identify structural load paths and evaluate the efficiency of the housing structural design (stiffness/weight).

PROGRAM INPUT AND OUTPUT FORMAT

The input format to S-68 is compatible with D-82 input format except as noted.

1. Cards after loader locations 0026 must be removed up to the "88888" card (sheet 1).

2. C-51 cards must be removed after the "99999" card following the masses input.

3. At end of deck, add number of modes to be analyzed (I5, card 1) and mode shapes desired (16I5, card 2).

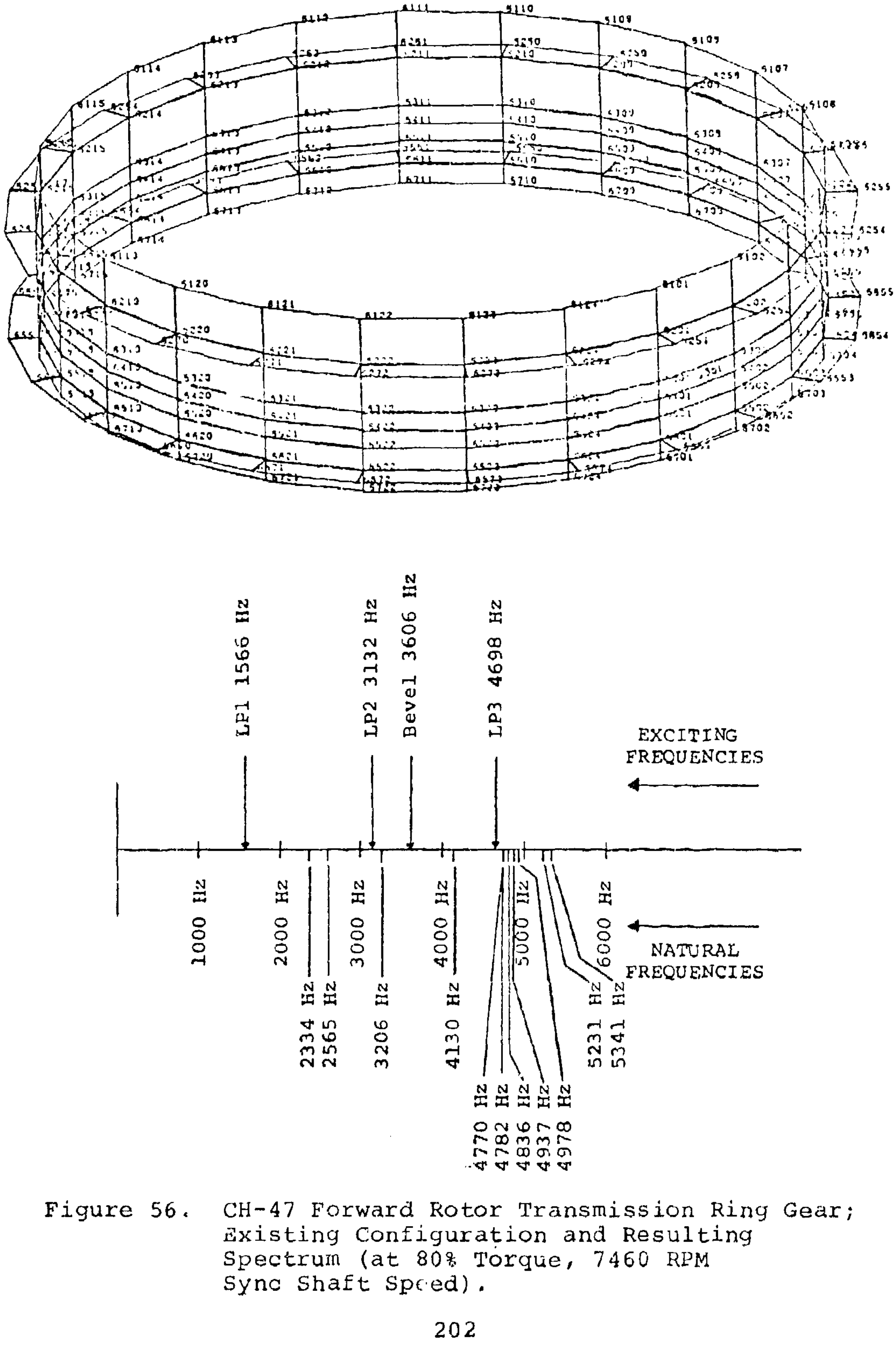


Figure 56. CH-47 Forward Rotor Transmission Ring Gear; Existing Configuration and Resulting Spectrum (at 80% Torque, 7460 RPM Sync Shaft Speed).

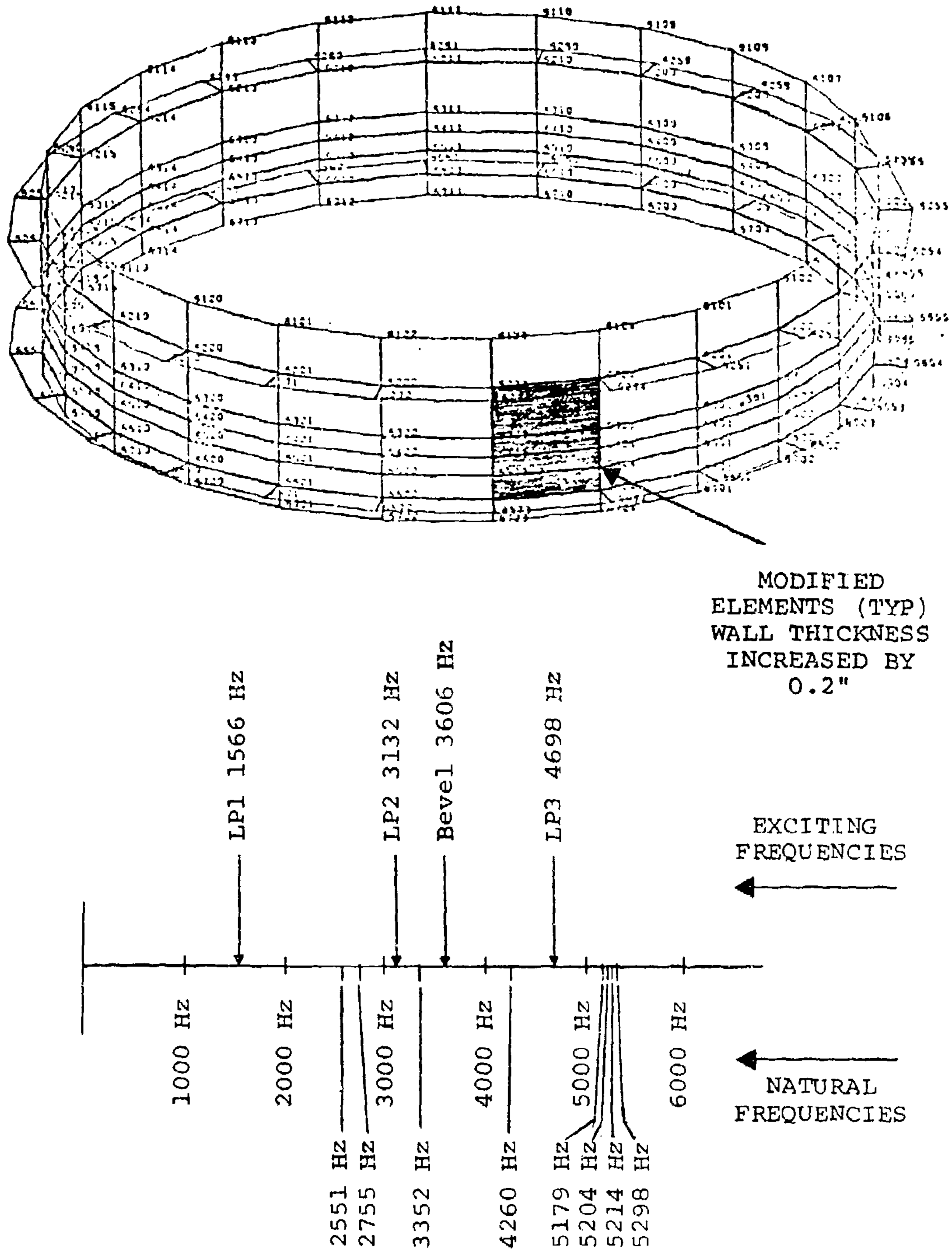


Figure 57. CH-47 Forward Rotor Transmission Ring Gear; Modified Configuration and Resulting Spectrum (at 80% Torque, 7460 RPM Sync Shaft Speed).

An example of output from the strain energy density computer program S-68 is shown in Figure 58. In addition, a flow diagram of the operation of computer program S-68 is shown in Figure 59. Figure 60 illustrates the application of computer program S-68 to the CH-47C forward pylon natural frequencies alteration. This example shows how modification to the structure can move the natural frequencies away from an exciting frequency and thereby reduce the amplification factor.

Since S-83 will calculate centerline stresses from data containing bending stresses, a flag must be input to indicate whether or not bending forces are to be processed and, if not, which of the two fiber distances will be used. Output includes a reference plate thickness, and when CTRIA1 or CQUAD1 elements exist, the user must flag the thickness to be output; the default is membrane thickness. The thickness specified has no effect on the output.

Figure 61 shows a typical S-83 input sheet which is rather standard except for Poisson's ratio and the modulus of elasticity. The cards are divided into ten fields of eight columns each. Figure 62 shows a typical S-83 strain density output (sorted) for the CH-47C forward transmission housing. Using this output to identify high strain density areas, a compromise modification (two contour plates) to the housing for detuning natural frequencies near four mesh excitions is shown in Figure 63.

CARD 1

FIELD

Col	1-8	-	GPL	(LEFT JUSTIFIED)
Col	9-16	-	EST	(LEFT JUSTIFIED)
Col	17-24	-	ØES1	(LEFT JUSTIFIED)
Col	25-32	-	ØEF1	(LEFT JUSTIFIED)
Col	33-40	-	BENDING OR MEMBRANE	(LEFT JUSTIFIED)
Col	41-48	-	FORCE FLAG	(RIGHT JUSTIFIED INTEGER)
Col	57-64	-	MODE OPTION	(LEFT JUSTIFIED)
Col	65-72	-	e-Young's Modulus	(REAL)
Col	73-80	-	u-Poisson's Ratio	(REAL)

```
CH-47C  XMSN SYSTEM COUPLED SCIARRA .5 PCSI 122 DOF
```

Natural Frequency = 1526 Hz

```
**** MODE SHAPE NO.  7 ****
```

Lower Planetary
First Harmonic = 1566 Hz

```
------------------------------------------------------------------------------------
                          AXIAL ELEMENT
N1= 135  N2= 335  MATERIAL CODE= 1  AREA= 0.203000 01
                             STRAIN ENERGY =   0.4405233D-02
 EL. VOL. / GROUP VCL.=  0.3396471D-C1  EL. VOL. / TOTAL VOL.=   0.3396471D-01
------------------------------------------------------------------------------------
                          AXIAL ELEMENT
N1= 125  N2= 325  MATERIAL CODE= 1  AREA= 0.780000 00
                             STRAIN ENERGY =   C.8874001D-03
 EL. VOL. / GROUP VCL.=  0.1305048D-C1  EL. VOL. / TOTAL VOL.=   0.1305048D-01
------------------------------------------------------------------------------------
                          BEAM ELEMENT
N1= 125  N2= 225  N3= 325  MATERIAL CODE= 1  IY= 0.100000-01  JX= 0.036000 02
AW= 0.100000-01          STRAIN ENERGY =   C.1777084D-03
 EL. VOL. / GROUP VCL.=  0.1496312D-C4  EL. VOL. / TOTAL VOL.=   0.1073138D-03
------------------------------------------------------------------------------------
                          AXIAL ELEMENT
N1= 435  N2= 535  MATERIAL CODE= 1  AREA= C.206000 01
                             STRAIN ENERGY =   0.1539410D-03
 EL. VOL. / GROUP VCL.=  0.3446665D-C1  EL. VOL. / TOTAL VOL.=   0.3446665D-01
------------------------------------------------------------------------------------
                          BEAM ELEMENT
N1= 135  N2= 137  N3= 335  MATERIAL CODE= 1  IY= 0.130000 02  JX= 0.0
AW= 0.560000 01          STRAIN ENERGY =   C.1351969D-03
 EL. VOL. / GROUP VCL.=  0.6284197D-C3  EL. VOL. / TOTAL VCL.=   0.7026830D-02
------------------------------------------------------------------------------------
                          BEAM ELEMENT
N1= 113  N2= 115  N3= 325  MATERIAL CODE= 1  IY= 0.120000 01  JX= 0.0
AW= 0.260000 01          STRAIN ENERGY =   C.1161701D-03
 EL. VOL. / GROUP VCL.=  0.3695706D-C3  EL. VOL. / TOTAL VOL.=   0.4132445D-02
------------------------------------------------------------------------------------
                          BEAM ELEMENT
N1= 111  N2= 113  N3= 325  MATERIAL CODE= 1  IY= 0.500000 00  JX= 0.0
AW= 0.130000 01          STRAIN ENERGY =   C.7512387D-04
 EL. VOL. / GROUP VOL.=  0.6807880D-C4  EL. VOL. / TOTAL VOL.=   0.7612399D-03
------------------------------------------------------------------------------------
                          BEAM ELEMENT
N1= 109  N2= 111  N3= 325  MATERIAL CODE= 1  IY= 0.170000 01  JX= 0.0
AW= 0.330000 01          STRAIN ENERGY =   C.6769816D-04
 EL. VOL. / GROUP VCL.=  0.9875166D-C3  EL. VOL. / TOTAL VCL.=   0.1104216D-01
------------------------------------------------------------------------------------
                          BEAM ELEMENT
N1= 128  N2= 129  N3= 325  MATERIAL CODE= 1  IY= 0.570000 01  JX= 0.0
AW= 0.270000 01          STRAIN ENERGY =   C.5568309D-04
 EL. VOL. / GROUP VCL.=  0.5655777D-C3  EL. VOL. / TOTAL VOL.=   0.6324147D-02
------------------------------------------------------------------------------------
                          AXIAL ELEMENT
N1= 545  N2= 547  MATERIAL CODE= 1  AREA= C.170000 01
                             STRAIN ENERGY =   C.5262069D-04
 EL. VOL. / GROUP VCL.=  0.2844193D-C1  EL. VOL. / TOTAL VOL.=   0.2844193D-01
------------------------------------------------------------------------------------
                          AXIAL ELEMENT
N1= 546  N2= 548  MATERIAL CODE= 1  AREA= 0.170000 01
                             STRAIN ENERGY =   C.5282053D-04
 EL. VOL. / GROUP VCL.=  0.2844193D-C1  EL. VOL. / TOTAL VOL.=   0.2844193D-01
```

Figure 58. Example of Strain Density Output (S-68).

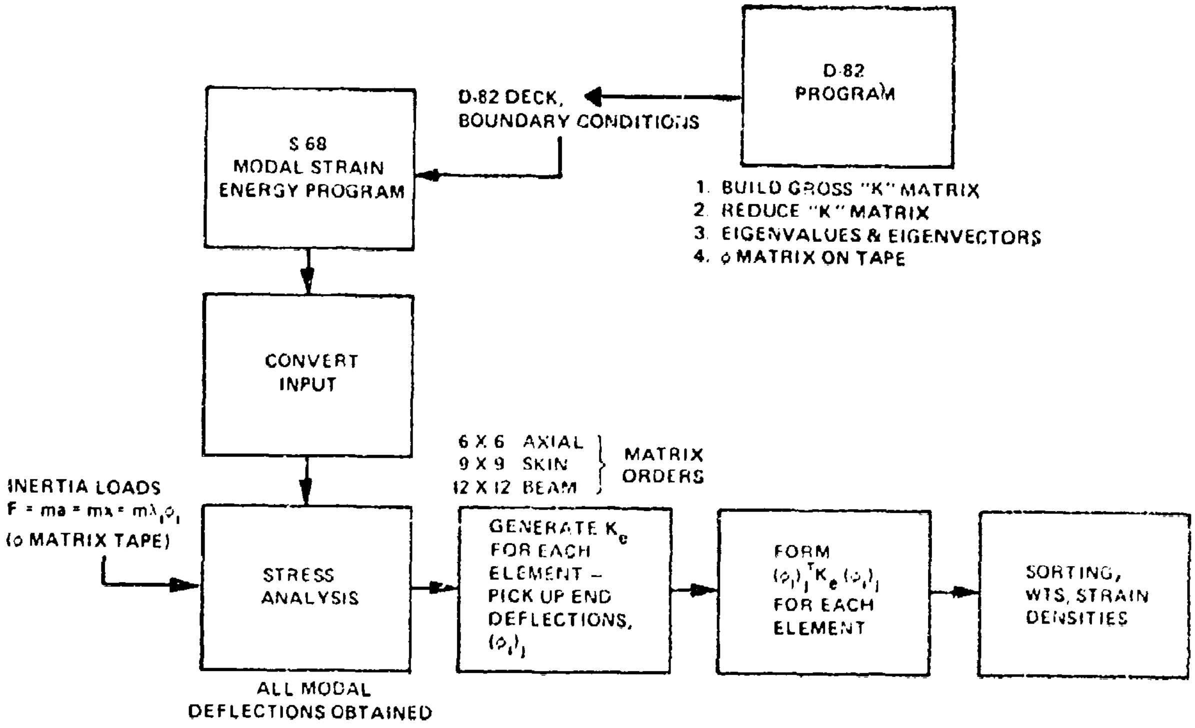


Figure 59. S-68 Modal Strain Energy Program Flow Diagram.

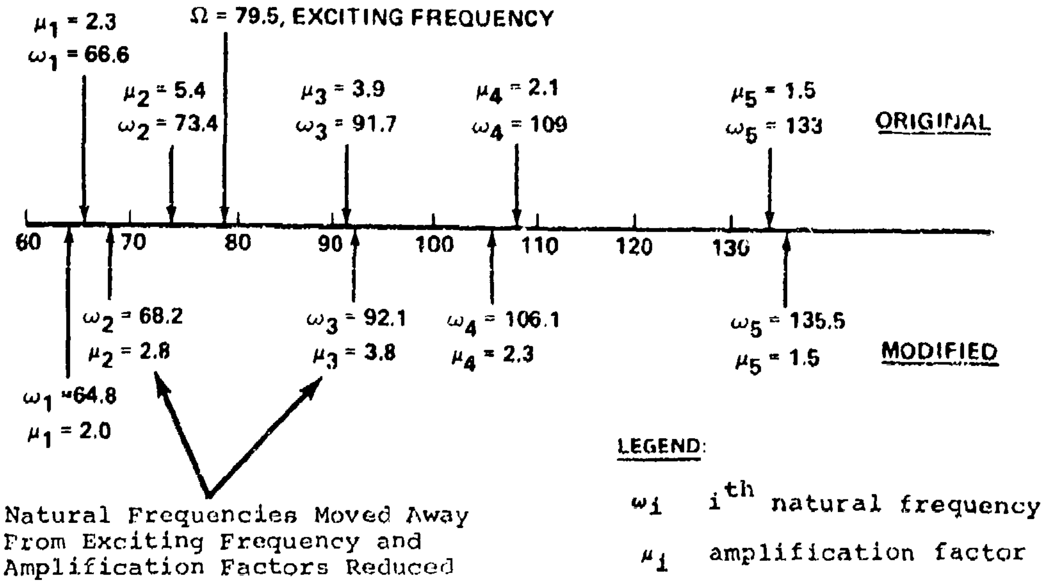


Figure 60. Example of Optimization of Natural Frequency Spectrum, CH-47 Helicopter Fuselage Forward Pylon Structure.

PROJECT INPUT S-83 | NAME

PROBLEM | DATE | PAGE OF

FIELD 1	FIELD 2	FIELD 3	FIELD 4	FIELD 5	FIELD 6	FIELD 7	FIELD 8	FIELD 9	FIELD 10
GPL	EST	DESI	DEFL	BENDING	-1		MØDE	10.556	.3125
$ØMIT	INCLUD	=	$END						

FORM 83000 (6/73)

Figure 61. Sample Input Sheet S-83.

ELEMENT ID	PRINCIPAL STRESSES MAJOR	PRINCIPAL STRESSES MINOR	STRAIN DENSITY
6317	-.2839416E 07	-.3863717E 07	0.269813E 06
6316	-.2835387E 07	-.3863722E 07	0.269811E 06
6315	-.2839112E 07	-.3863575E 07	0.269793E 06
6314	-.2834259E 07	-.3863558E 07	0.269752E 06
6318	-.2839227E 07	-.3863446E 07	0.269775E 06
6319	-.2839100E 07	-.3863273E 07	0.269751E 06
6313	-.2839044E 07	-.3863203E 07	0.269741E 06
6312	-.2832954E 07	-.3863027E 07	0.269719E 06
6311	-.2838827E 07	-.3862694E 07	0.269658E 06
6310	-.283[illegible]757E 07	-.3862792E 07	0.269685E 06
6320	-.2838721E 07	-.3862782E 07	0.269662E 06
6309	-.2838662E 07	-.3862459E 07	0.269670E 06
6308	-.2838556E 07	-.3862555E 07	0.269653E 06
6321	-.2838487E 07	-.3862414E 07	0.269632E 06
6307	-.2838423E 07	-.3862362E 07	0.269626E 06
6306	-.2838352E 07	-.3862273E 07	0.269606E 06
6305	-.2838207E 07	-.3862078E 07	0.269584E 06
6322	-.2838056E 07	-.3861935E 07	0.269563E 06
6304	-.2838100E 07	-.3861503E 07	0.269560E 06
6303	-.2837918E 07	-.3861665E 07	0.269527E 06
6323	-.2837900E 07	-.3861521E 07	0.269511E 06
6302	-.2837795E 07	-.3861444E 07	0.269498E 06
6324	-.2837613E 07	-.3861204E 07	0.269464E 06
6301	-.2837624E 07	-.3861177E 07	0.269462E 06
6267	-.1233874E 07	-.4198710E 07	0.265661E 06
6266	-.1233805E 07	-.4198757E 07	0.265658E 06
6265	-.1233003E 07	-.4198595E 07	0.265645E 06
6268	-.1233605E 07	-.4198453E 07	0.265628E 06
6264	-.1233732E 07	-.4198428E 07	0.265624E 06
6269	-.1233709E 07	-.4198705E 07	0.265596E 06
6263	-.1233668E 07	-.4198164E 07	0.265590E 06
6262	-.1233584E 07	-.4197564E 07	0.265565E 06
6261	-.1233624E 07	-.4197832E 07	0.265549E 06
6260	-.1233481E 07	-.4197713E 07	0.265536E 06
6259	-.1233507E 07	-.4197650E 07	0.265525E 06
6270	-.1233434E 07	-.4197626E 07	0.265523E 06
6258	-.1233394E 07	-.4197424E 07	0.265502E 06
6271	-.1233484E 07	-.4197352E 07	0.265489E 06
6257	-.1233475E 07	-.4197277E 07	0.265478E 06
6256	-.1233284E 07	-.4197070E 07	0.265452E 06
6255	-.1233365E 07	-.4196525E 07	0.265433E 06
6272	-.1233240E 07	-.4196806E 07	0.265419E 06
6254	-.1233195E 07	-.4196755E 07	0.265413E 06
6253	-.1233222E 07	-.4196530E 07	0.265384E 06
6273	-.1233156E 07	-.4196474E 07	0.265371E 06
6252	-.1233100E 07	-.4196300E 07	0.265355E 06
6274	-.1233128E 07	-.4196044E 07	0.265322E 06
6251	-.1233031E 07	-.4196013E 07	0.265318E 06
6667	0.2312830E 07	0.5978177E 06	0.808221E 05
6666	0.2312800E 07	0.5977836E 06	0.808200E 05
6665	0.2312748E 07	0.5977733E 06	0.808164E 05
6668	0.2312726E 07	0.5977199E 06	0.808152E 05
6664	0.2312661E 07	0.5977530E 06	0.808103E 05
6669	0.2312577E 07	0.5977044E 06	0.808043E 05
6663	0.2312550E 07	0.5977260E 06	[illegible] 05
6662	0.2312453E 07	0.5976529E 06	[illegible] 05
6661	0.2312377E 07	0.5976513E 06	0.807503E 05
6670	0.2312258E 07	0.5976357E 06	0.807450E 05
6660	0.2312289E 07	0.5976361E 06	0.807444E 05
6659	0.2312229E 07	0.5976566E 06	0.807401E 05
6658	0.2312137E 07	0.5975566E 06	0.807730E 05
6657	0.2312065E 07	0.5976173E 06	0.807687E 05
6671	0.2312044E 07	0.5976013E 06	0.807672E 05
6656	0.2311965E 07	0.5975547E 06	0.807620E 05
6655	0.2311877E 07	0.5975661E 06	0.807554E 05
6672	0.2311771E 07	0.5975225E 06	0.807481E 05
6654	0.2311759E 07	0.5975062E 06	0.807473E 05
6653	0.2311629E 07	0.5974863E 06	0.807382E 05
6673	0.2311548E 07	0.5974414E 06	0.807326E 05
6652	0.2311513E 07	0.5974564E 06	0.807301E 05
6674	0.2311427E 07	0.5974326E 06	0.807239E 05
6651	0.2311410E 07	0.5974115E 06	0.807229E 05
6116	0.9715977E 06	-.1690560E 07	0.803400E 05
6117	0.9716066E 06	-.1690544E 07	0.803398E 05
6118	0.9715129E 06	-.1690535E 07	0.803336E 05
6115	0.9715240E 06	-.1690520E 07	0.803331E 05
6114	0.9715838E 06	-.1690420E 07	0.803295E 05
6119	0.9716184E 06	-.1690263E 07	0.803208E 05
6113	0.9714267E 06	-.1690327E 07	0.803155E 05
6112	0.9714470E 06	-.1690270E 07	0.803119E 05
6111	0.9713807E 06	-.1690191E 07	0.803091E 05
6110	0.9714009E 06	-.1690145E 07	0.803021E 05
6120	0.9712488E 06	-.1690253E 07	0.803017E 05
6109	0.9713166E 06	-.1690115E 07	0.802959E 05
6108	0.9713674E 06	-.1690046E 07	0.802938E 05
6121	0.9714037E 06	-.1685864E 07	0.802836E 05
6107	0.9712254E 06	-.1683960E 07	0.802811E 05
6106	0.9712775E 06	-.1683878E 07	0.802783E 05
6105	0.9711603E 06	-.1685839E 07	0.802698E 05
6104	0.9712944E 06	-.1684753E 07	0.802663E 05
6122	0.9710257E 06	-.1685862E 07	0.802649E 05
6103	0.9710259E 06	-.1686674E 07	0.802524E 05

Figure 62. S-83 Strain Density Output (4770 Hz).

6123	0.9712073E 06	-.1689502E 07	0.802496E 05
6102	0.9711347E 06	-.1689533E 07	0.802483E 05
6124	0.9705162E 06	-.1689463E 07	0.802127E 05
6101	0.9705511E 06	-.1689356E 07	0.802300E 05
6516	0.1666465E 07	0.7373C1E 05	0.452161E 05
6517	0.1665451E 07	0.7365561E 05	0.452153E 05
6515	0.1666414E 07	0.7365636E 05	0.452131E 05
6514	0.1666348E 07	0.7363409E 05	0.452099E 05
6518	0.1666343E 07	0.7365588E 05	0.452049E 05
6513	0.1666269E 07	0.7365275E 05	0.452055E 05
6519	0.1666264E 07	0.7367663E 05	0.452054E 05
6512	0.1666152E 07	0.7367736E 05	0.452015E 05
6511	0.1666141E 07	0.7366700E 05	0.451985E 05
6520	0.1666058E 07	0.7370413E 05	0.451959E 05
6510	0.1666064E 07	0.7366431E 05	0.451946E 05
6509	0.1666046E 07	0.7368375E 05	0.451933E 05
6508	0.1667964E 07	0.7366263E 05	0.451492E 05
6507	0.1667918E 07	0.7366106E 05	0.451864E 05
6521	0.1667857E 07	0.7363550E 05	0.451354E 05
6506	0.1667827E 07	0.7365606E 05	0.451318E 05
6505	0.1667787E 07	0.7367519E 05	0.451793E 05
6522	0.1667725E 07	0.7368244E 05	0.451758E 05
6504	0.1665758E 07	0.7366044E 05	0.451[illegible]E 05
6503	0.1665760E 07	0.7365025E 05	0.451694E 05
6523	0.1665753E 07	0.7361150E 05	0.451662E 05
6502	0.1667505E 07	0.7364031E 05	0.451643E 05
6524	0.1667506E 07	0.7367500E 05	0.451638E 05
6501	0.1667437E 07	0.7363550E 05	0.451607E 05
6416	0.7083128E 06	-.1075474E 07	0.355108E 05
6417	0.7082826E 06	-.1075461E 07	0.355095E 05
6415	0.7082628E 06	-.1075454E 07	0.355082E 05
6414	0.7082651E 06	-.1075413E 07	0.355065E 05
6418	0.7082474E 06	-.1075426E 07	0.355065E 05
6419	0.7082595E 06	-.1075343E 07	0.355033E 05
6413	0.7082013E 06	-.1075355E 07	0.355023E 05
6412	0.7082202E 06	-.1075313E 07	0.355006E 05
6411	0.7081777E 06	-.1075272E 07	0.354974E 05
6420	0.7081556E 06	-.1075227E 07	0.354948E 05
6410	0.7081311E 06	-.1075250E 07	0.354948E 05
6409	0.7081273E 06	-.1075215E 07	0.354932E 05
6408	0.7080653E 06	-.1075177E 07	0.354902E 05
6407	0.7080F06E 06	-.1075130E 07	0.354879E 05
6421	0.7080600E 06	-.1075140E 07	0.354876E 05
6406	0.7080359E 06	-.1075064E 07	0.354845E 05
6405	0.7080227E 06	-.1075030E 07	0.354819E 05
6422	0.7075634E 06	-.1075021E 07	0.354791E 05
6404	0.7075754E 06	-.1074594E 07	0.354784E 05
6403	0.7079276E 06	-.1074540E 07	0.354744E 05
6423	0.7075164E 06	-.1074532E 07	0.354737E 05
6402	0.7079052E 06	-.1074655E 07	0.354712E 05
6424	0.7073933E 06	-.1074635E 07	0.354687E 05
6401	0.7078734E 06	-.1074645E 07	0.354686E 05
6217	0.6276854E 06	-.8294076E 06	0.234115E 05
6216	0.6277154E 06	-.8293741E 06	0.234114E 05
6215	0.6277306E 06	-.8293567E 06	0.234110E 05
6214	0.6277116E 06	-.8293246E 06	0.234093E 05
6218	0.6276752E 06	-.8293469E 06	0.234091E 05
6213	0.6276969E 06	-.8292664E 06	0.234067E 05
6219	0.6275914E 06	-.8293372E 06	0.234062E 05
6212	0.6276754E 06	-.8292316E 06	0.234051E 05
6211	0.6276674E 06	-.8292002E 06	0.234036E 05
6220	0.6276710E 06	-.8291415E 06	0.234019E 05
6210	0.6276276E 06	-.8291765E 06	0.234018E 05
6209	0.6276062E 06	-.8291567E 06	0.234005E 05
6208	0.6275635E 06	-.8291416E 06	0.233993E 05
6221	0.6275052E 06	-.8291323E 06	0.233966E 05
6207	0.6275578E 06	-.8290635E 06	0.233954E 05
6206	0.6275286E 06	-.8290235E 06	0.233950E 05
6205	0.6275049E 06	-.8290295E 06	0.233932E 05
6222	0.6275708E 06	-.8269375E 06	0.233914E 05
6204	0.6274723E 06	-.8269604E 06	0.233908E 05
6223	0.6274564E 06	-.8269499E 06	0.233690E 05
6203	0.6274458E 06	-.8269330E 06	0.233882E 05
6202	0.6274426E 06	-.8286914E 06	0.233860E 05
6201	0.6274566E 06	-.8288218E 06	0.233846E 05
6224	0.6275204E 06	-.8287645E 06	0.233846E 05
6617	0.3461461E 06	0.1221094E 06	0.180873E 04
6616	0.3461366E 06	0.1221023E 06	0.180860E 04
6615	0.3461254E 06	0.1221089E 06	0.180848E 04
6614	0.3461171E 06	0.1220893E 06	0.180839E 04
6618	0.3461115E 06	0.1221455E 06	0.180836E 04
6619	0.3460983E 06	0.1220599E 06	0.180818E 04
6613	0.3460823E 06	0.1220544E 06	0.1[illegible]03E 04
6612	0.3460714E 06	0.1220653E 06	0.180791E 04
6610	0.3460551E 06	0.1220747E 06	0.180778E 04
6611	0.3460523E 06	0.1220753E 06	0.180772E 04
6609	0.3460432E 06	0.1220791E 06	0.180767E 04
6608	0.3460373E 06	0.1220585E 06	0.180755E 04
6621	0.3460360E 06	0.1220335E 06	0.180753E 04
6620	0.3460255E 06	0.1220695E 06	0.180749E 04
6607	0.3460105E 06	0.1220641E 06	0.180726E 04
6606	0.3459978E 06	0.1220438E 06	0.180714E 04
6605	0.3459793E 06	0.1220545E 06	0.180696E 04
6604	0.3459601E 06	0.1220333E 06	0.180695E 04
6622	0.3459757E 06	0.1220652E 06	0.180693E 04
6603	0.3459518E 06	0.1220423E 06	0.180668E 04
6623	0.3459474E 06	0.1220172E 06	0.180661E 04
6602	0.3459413E 06	0.1220313E 06	0.180655E 04
6624	0.3458584E 06	0.1215883E 06	0.180409E 04
6601	0.3458919E 06	0.1220109E 06	0.180604E 04

Figure 62. Continued.

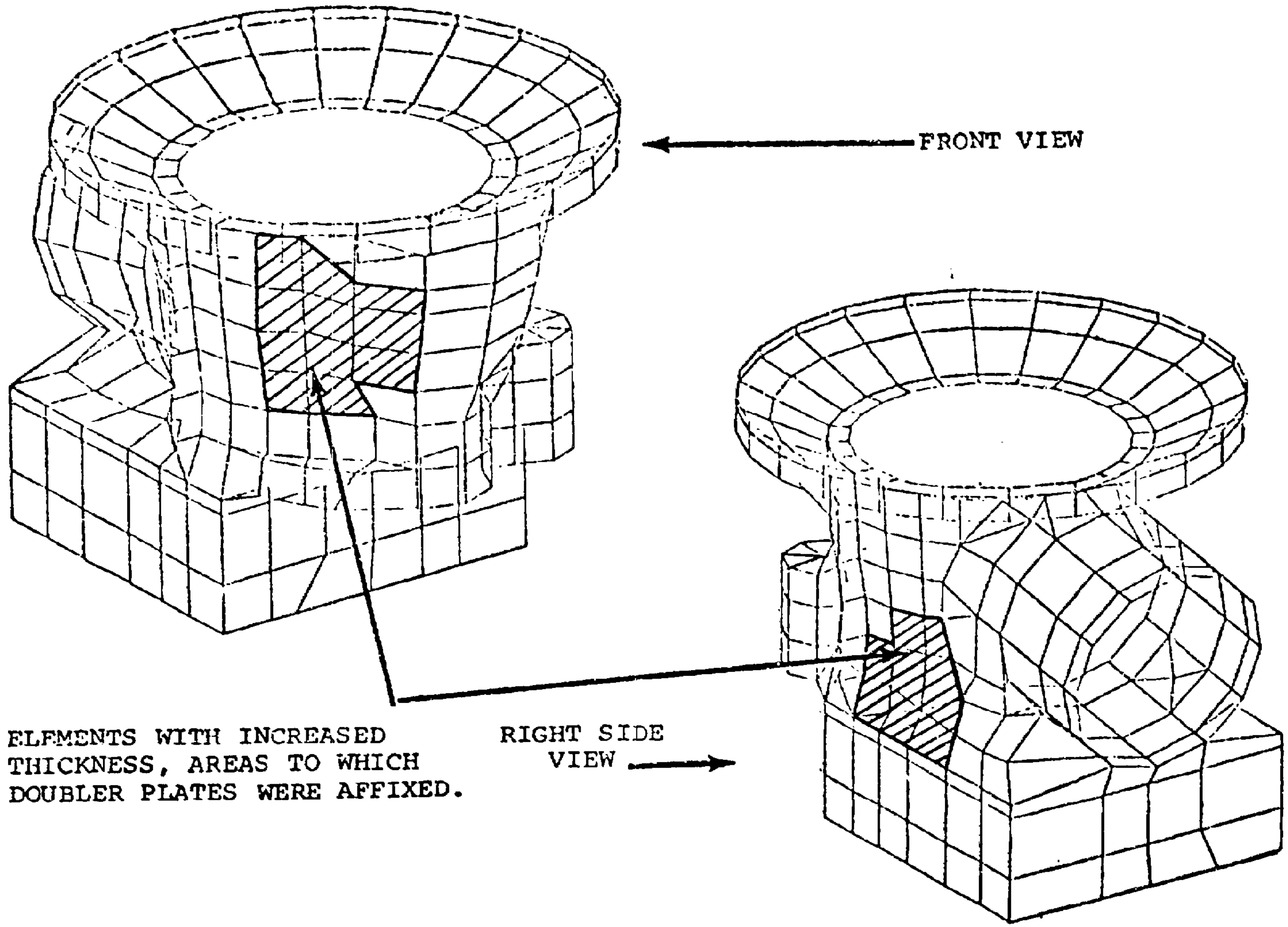


Figure 63. CH-47 Forward Transmission Case With Modifications (Crosshatched Areas) to Wall Thickness.

NOTES:

1. Mnemonics are "bending" or "membrane".

2. Force Flag: Process plate forces, calculate centerline stresses

   =-1, Do not process plate forces, use first fibre stress

   =-2, Do not process plate forces, use second fibre stress

<u>CARD 2</u>

```
COLUMN 2
|
↓
$OMIT          IXCLUD =   $END
```

Program S-83 includes strain density calculations for CQUAD1, CTRIA2, CQUAD2 and CTRIA2 elements.

CONCLUSIONS

A complex gearbox such as a helicopter rotor transmission typically has more than one gear mesh, hence more than one exciting frequency. For instance, the Boeing Vertol CH-47C helicopter forward rotor transmission employs a spiral bevel gear mesh plus a two-stage planetary gear system. Additional sources of exciting frequencies in the form of sidebands are introduced by planetary gear configurations and manufacturing variations. This occurrence of multiple exciting frequencies, coupled with the fact that the housing possesses many natural frequencies, makes it a complex task to detune the housing so that none of the exciting frequencies coincides with a natural frequency. The primary frequencies for the CH-47 forward rotor transmission have been identified experimentally as the bevel gear mesh frequency and the lower planetary gear mesh frequency (LP1) and its second (LP2) and third (LP3) harmonics.

Experimentation which included the dynamic testing of a CH-47C forward transmission with internal instrumentation to measure strains, displacements, and accelerations of the rotating components and external instrumentation to measure housing acceleration and noise indicated that by modifying the gear/shaft/bearing system geometry the internal components may be detuned to minimize excitation of the housing. Application of strain density techniques to these dynamic components has identified modifications which have analytically reduced the loads exciting the housing at the bevel mesh, LP2 and LP3 frequencies, but increased the loads at the LP1 frequency. Since the effects of multiple noise sources are added logarithmically, the reduction of three out of four noise sources may not appreciably reduce the overall noise level.

Noise measurements have tended to confirm that housing responses exist and generate noise. This is evidenced, for example, by the LP2 and LP3 frequencies. Although the exciting source for these frequencies is within the ring gear, the maximum noise at these frequencies emanates from the mid-case region.

It is important to note that since the exciting frequencies will vary with changes in operating speed, the housing must be detuned at a specific operation speed (±3%). The use of strain density has led to preliminary identification of the areas of the housing structure which must be modified to detune the housing for reduced vibration/noise. The strain density distribution was determined using the NASTRAN post-processor for the modes with frequencies nearest to the four main exciting frequencies, and the elements with high strain density were identified. For each mode considered, the elements with high strain density are generally different; however, some elements are common to two or more of the modes. Strictly speaking, the elements with highest strain density for each

mode should be modified to achieve the maximum frequency shift for each corresponding mode. This approach would be used during the design of a new housing. To modify an existing housing, however, it would be cumbersome to incorporate the numerous and varied modifications indicated by such a rigorous application of the analysis. Therefore, for practical application to the experimental housing herein, those elements with a relatively high strain density which are common to two or more modes have been identified and were used to shift the housing frequencies. In this manner a specified structural change altered two or more frequencies, although perhaps no single frequency would be shifted maximally. It is more feasible to modify these elements, since the actual changes to the existing housing design for testing were limited to a few easily accessible areas on the exterior walls of the housing. This approach provided sufficient detuning to demonstrate the validity of the analyses. Prior to finalizing the detuned design, the dynamic response of the model, with the structural modification incorporated, was recalculated using NASTRAN.

The basic analytical approach used as a design tool for transmission vibration/noise reduction has been partially validated. The method unites the internal components and the housing, and hence optimizes the transmission as a complete operating system. Since the housing provides structural support to the internal components, its physical characteristics grossly affect performance and life in terms of internal bearing capacity, gear capacity, fretting, misalignments, etc. Therefore, housing optimization is essential if the full benefit of the advancements in gear and bearing technology are to be realized.

REFERENCES

1. Badgley, R.H., and Laskin, I., PROGRAM FOR HELICOPTER GEARBOX NOISE PREDICTION AND REDUCTION, Mechanical Technology Incorporated, USAAVLABS Technical Report 70-12, U.S. Army Aviation Materiel Laboratories, Fort Eustis, Virginia, March 1970, AD869822.

2. Laskin, I., Orcutt, F.K., and Shipley, E.E., ANALYSIS OF NOISE GENERATED BY UH-1 HELICOPTER TRANSMISSION, Mechanical Technology Incorporated, USAAVLABS Technical Report 68-41, U.S. Army Aviation Materiel Laboratories, Fort Eustis, Virginia, June 1968, AD675457.

3. Hartman, R.M., A DYNAMICS APPROACH TO HELICOPTER TRANSMISSION NOISE REDUCTION AND IMPROVED RELIABILITY, Presented at the 29th Annual National Forum of the American Helicopter Society, Washington, D.C., May 1973.

4. Hartman, R.M., and Badgley, R.H., MODEL 301 HLH/ATC TRANSMISSION NOISE REDUCTION PROGRAM, The Boeing Vertol Company, USAAMRDL Technical Report 74-58, U.S. Army Air Mobility Research and Development Laboratory, Fort Eustis, Virginia, May 1974, AD784132.

5. Sciarra, J.J., VIBRATION REDUCTION BY USING BOTH THE FINITE ELEMENT STRAIN ENERGY DISTRIBUTION AND MOBILITY TECHNIQUES, 45th Shock and Vibration Symposium, Dayton, Ohio, August 1974.

6. Sciarra, J.J., USE OF THE FINITE ELEMENT DAMPED FORCED RESPONSE STRAIN ENERGY DISTRIBUTION FOR VIBRATION REDUCTION, U.S. Army Research Office - Durham, Final Report Contract DAH-CO4-71-C-0048, July 1974.

7. Gu, A.L., and Badgley, R.H., PREDICTION OF GEAR-MESH-INDUCED HIGH-FREQUENCY VIBRATION SPECTRA IN GEARED POWER TRAINS, Mechanical Technology Incorporated, USAAMRDL Technical Report 74-5, U.S. Army Air Mobility Research and Development Laboratory, Fort Eustis, Virginia, January 1974, AD777496.

8. Badgley, R.H., MECHANICAL ASPECTS OF GEAR-INDUCED NOISE IN COMPLETE POWER TRAIN SYSTEMS, ASME Paper No. 70-WA/DGP-1, Presented at the ASME Winter Annual Meeting, New York, December 1970.

9. Badgley, R.H., GEARBOX DYNAMICS -- THE KEY TO UNDERSTANDING AND REDUCING ACOUSTIC-FREQUENCY ENERGY IN GEARED POWER TRAINS, Presented at the Meeting of the Aerospace Gearing Committee of the American Gear Manufacturers Association, Cleveland, Ohio, January 17-18, 1972.

10. Badgley, R.H., and Chiang, T., INVESTIGATION OF GEARBOX DESIGN MODIFICATIONS FOR REDUCING HELICOPTER GEARBOX NOISE, Mechanical Technology Incorporated, USAAMRDL Technical Report 72-6, U.S. Army Mobility Research and Development Laboratory, Fort Eustis, Virginia, March 1972, AD742735.

11. Badgley, R.H., REDUCTION OF NOISE AND ACOUSTIC-FREQUENCY VIBRATIONS IN AIRCRAFT TRANSMISSIONS, AHS Paper No. 661, Presented at the 28th Annual National Forum of the American Helicopter Society of Washington, D.C., May 1972.

12. Badgley, R.H., and Hartman, R.M., GEARBOX NOISE REDUCTION: PREDICTION AND MEASUREMENT OF MESH-FREQUENCY VIBRATIONS WITHIN AN OPERATING HELICOPTER ROTOR-DRIVE GEARBOX, ASME Paper No. 73-DET-31, Presented at the ASME Design Engineering Technical Conference, Cincinnati, Ohio, September 9-12, 1973.

13. Chiang, T., and Badgley, R.H., REDUCTION OF VIBRATION AND NOISE GENERATED BY PLANETARY RING GEARS IN HELICOPTER AIRCRAFT TRANSMISSIONS, ASME Paper No. 72-PTG-11, Presented at ASME Mechanisms Conference and International Symposium on Gearing and Transmissions, San Francisco, California, October 8-12, 1972.

14. Gu, A.L., and Badgley, R.H., PREDICTION OF VIBRATION SIDEBANDS IN GEAR MESHES, ASME Paper 74-DET-95, Presented at the Design Engineering Technical Conference, New York, New York, October 5-9, 1974.

15. O'Donnel, W.J., STRESS AND DEFLECTION IN BUILT-IN BEAMS, ASME Paper No. 62-WA-16, 1962.

16. Palmgren, David, BALL AND ROLLER BEARING ENGINEERING, 3rd Edition, SKF Industries, Inc.

17. Sciarra, J.J., A COMPUTER METHOD FOR DYNAMIC STRUCTURAL ANALYSIS USING STIFFNESS MATRIXES, Journal of Aircraft, Vol. 6, No. 1, January - February 1969.

18. Sciarra, J.J., USE OF THE FINITE ELEMENT DAMPED FORCED RESPONSE STRAIN ENERGY DISTRIBUTION FOR VIBRATION REDUCTION, Presented at the ARO-D Military Theme Review, The Helicopter and V/STOL Aircraft Research Conference, Moffett Field, California, September 1972.

19. Jones, A.B., A GENERAL THEORY FOR ELASTICALLY CONSTRAINED BALL AND RADIAL ROLLER BEARINGS UNDER ARBITRARY LOAD AND SPEED CONDITIONS, ASME Paper 59-LUB-10.

20. Reed, D.L., POINT STRESS LAMINATE ANALYSIS, Document FZM-5494, Prepared for Advanced Composite Division, Air Force Materials Laboratory, WPAFB, Ohio, April 1970.

21. Everstine, G., BANDIT - A COMPUTER PROGRAM TO RENUMBER NASTRAN GRID POINTS FOR REDUCED BANDWIDTH, Naval Ship Research and Development Center Technical Note AML-6-70, February 1970.

APPENDIX A

LISTING OF GEAR TOOTH MESH EXCITATION AND COMPLIANCE INPUT GENERATOR PROGRAM FOR R-75 (GGEAR/HCR) OR R-67 (GGEAR)

```
      •JOB              R.DRAGO,KP=29,PAGES=250,LINES=60,RUN=CHECK
            DIMENSION RB(10),YE(10),GE(10),POS(10),VPT(10),IDENT(20),FN(10)
            DIMENSION RP(10),RO(10),RT(10),ARM(10),AT(10),F(10),RF(10)
            DIMENSION UJ1(80),UJ2(80),ZJ1(80),ZJ2(80),TOR(80)
            E=30.E6
            ANU=0.3
            NWS = 1
            DEGTOR = 0.017453
      1999  I = 1
            IRM = 0
            IPERC = 0
            IRF = 1
            IPROP = 0
            ICENT = 0
            IDIAG = 0
            IHICR=0
            XINT=1.0
            NTS=1
            READ(5,4004) IDENT
      4004  FORMAT(20A4)
C
C
C           TREGOLD APPROXIMATION OF A SPUR GEAR FROM A BEVEL OR HELICAL GEAR
C
C           INPUT DATA GENERATOR
C           PROVIDES PUNCHED DECK FOR DIRECT INPUT TO GEARO,GGEAR,OR GGEAR/HCR
C
C            * * * * * * * * * *   INPUT DATA  * * * * * * * * * * *
C
C           CARD # 1  -  TITLE
C
C           CARD #2
C                 J      = OUTPUT SELECTION
C                          0 = END OF JOB,TERMINATE EXECUTION
C                          1 = TREGOLD ONLY
C                          2 = GEARO   (R-33)
C                          3 = GGEAR   (R-67)
C                          4 = GGEAR/HCR  (R-75)
C                            OR INPUT  J  AS NEGATIVE VALUE TO READ OPTIONAL
C                               MATERIAL PROPERTIES CARD
C                 TN1    = NO OF TEETH - DRIVING
C                 TN2    = NO OF TEETH - DRIVEN
C                 PCA1   = PITCH CONE ANGLE(DEGREES) - DRIVING
C                 PCA2   = PITCH CONE ANGLE(DEGREES) - DRIVEN
C                 SA     = SPIRAL OR HELIX ANGLE(DEGREES)
C                 FW1    = FACE WIDTH(INCHES) - DRIVING
C                 FW2    = FACE WIDTH(INCHES) - DRIVEN
C                 PD     = DIAMETRAL PITCH
C
C                 RO1    = OUTSIDE RADIUS(INCHES) - DRIVING
C                            OR IF INPUT AS A NEGATIVE VALUE
C                        = ADDENDUM(INCHES) - DRIVING
C
C                 RO2    = OUTSIDE RADIUS(INCHES) - DRIVEN
C                            OR IF INPUT AS A NEGATIVE VALUE
C                        = ADDENDUM(INCHES) - DRIVEN
C
C                 RR1    = ROOT RADIUS(INCHES) - DRIVING
C                            OR IF INPUT AS A NEGATIVE VALUE
C                        = DEDENDUM    (INCHES) - DRIVING
C
C                 RR2    = ROOT RADIUS(INCHES) - DRIVEN
C                            OR IF INPUT AS A NEGATIVE VALUE
C                        = DEDENDUM    (INCHES) - DRIVEN
```

```
C
C            T1   = CIRCULAR TOOTH THICK.(INCHES) - DRIVING
C            T2   = CIRCULAR TOOTH THICK.(INCHES) - DRIVEN
C                    OR IF T1 IS INPUT AS A NEGATIVE VALUE
C            T1   = PERCENT OF TOOTH THICK. SPLIT TO DRIVING MEMBER
C            T2   = BACKLASH(INCHES)
C
C            PA   = PRESSURE ANGLE(DEGREES)
C            FI   = NO OF CALCULATION POINTS
C            TSE1 = TOOTH SPACEING ERROR(INCHES) - DRIVING
C            TSE2 = TOOTH SPACEING ERROR(INCHES) - DRIVEN
C            MN   = 0 FOR INTERNAL, = 1 FOR EXTERNAL
C            MMM  = NO. OF FOURIER TERMS
C            TORK = TORQUE(IN-LB) - DRIVING (0=READ VARIABLE TORQUE
C                                           ON TOR(I) CARDS)
C            WS   = SPEED OF DRIVING MEMBER(RPM)
C            IPLT = PLOTTING CONTROL(0=NO, 1=YES)
C            DEV  = DEVIATION FROM STANDARD CENTER DISTANCE(INCHES)
C            NZJ  = SELECTOR FOR DEVIATION AND COMPLIANCE METHOD
C
C
C      CARD  -  IF NZJ = 0,4, OR 5   INSERT CARDS FOR PROFILE DEVIATIONS
C                                    DRIVING(UJ1) AND DRIVEN (UJ2)
C
C      CARD  -  IF NZJ = 0 OR 3      INSERT CARDS FOR TOOTH COMPLIANCE
C                                    DRIVING (ZJ1) AND DRIVEN (ZJ2)
C
C      CARD  -  IF TORK .LT. 0       INSERT CARDS FOR VARIABLE TORQUE
C                                    DRIVING (TOR)
C
C      CARD  -  IF NZJ = 1 OR 4      INSERT CARDS FOR TOOTH PROFILE
C                                    DEVIATION AT 4 POINTS ALONG PROFILE
C                                    DRIVING AND DRIVEN
C
C
C      CARD  -  IF J .LT. 0          INSERT MATERIAL PROPERTIES
C           YE(1) = MODULUS OF ELASTICITY - DRIVING
C           POS(1) = POISSONS RATIO - DRIVING
C           YE(2) = MODULUS OF ELASTICITY - DRIVEN
C           POS(2) = POISSONS RATIO - DRIVEN
C
C             * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C
  2000 READ,J,TN1,TN2,PCA1,PCA2,SA,FW1,FW2,' ),RO1,RO2,RR1,RR2,T1,
      XT2,PA    ,FI,TSE1,TSE2,MN,MMM,TORK,WS,IPLT,DEV,NZJ
       IF(J.EQ.0) GO TO 5556
       IF(J.LT.0) IPROP=1
       IF(IPROP.EQ.1) READ,YE(1),POS(1),YE(2),POS(2)
       J=IABS(J)
       N = 3
       IF(PCA1.EQ.0.AND.SA.GT.0) N = 2
       IF(SA.EQ.0) N = 1
       IF(MN.EQ.0) XINT=-1.0
       PAR=PA*DEGTOR
       IF(T1.GT.0.0) GO TO 1326
       CP=3.14159/PD-T2
       T1=ABS(T1*CP)
       T2=CP-T1
  1326 R1=TN1/(2*PD)
       R2=TN2/(2*PD)
       C=XINT*R1+R2+DEV
       IF(RO1.GT.0.) GO TO 1315
       A1=ABS(RO1)
       RO1=R1+A1
  1315 IF(RO2.GT.0.) GO TO 1316
       A2=ABS(RO2)
       RO2=R2+(A2*XINT)
  1316 IF(RR1.GT.0.) GO TO 1320
       B1=ABS(RR1)
       RR1=R1-B1
  1320 IF(RR2.GT.0.) GO TO 1325
       B2=ABS(RR2)
       RR2=R2-(B2*XINT)
       GO TO 1317
```

```
 1325 A1=RO1-R1
      A2=(RO2-R2)*XINT
      B1=R1-RR1
      B2=(R2-RR2)*XINT
 1317 D=A1+A2
      RB1=R1*COS(PAR)
      RB2=R2*COS(PAR)
      BASEP=2.*3.14159*RB1/TN1
      CR=(SQRT(RO1**2-RB1**2)+(SQRT(RO2**2-RB2**2)-C*SIN(PAR))*XINT)/
     1BASEP
      IF(CR.GE.2.0) IHICR=1
C
C TREGOLD APPROXIMATION OF A SPUR FROM A SPIRAL BEVEL GEAR
      TN = TN1
      PCA = PCA1
      FW = FW1
      R = R1
      A = A1
      B = B1
      T = T1
      TSE = TSE1
      XIN=1.0
      GO TO 6051
 6050 I = I+1
      TN = TN2
      PCA = PCA2
      FW = FW2
      R = R2
      A = A2
      B = B2
      T = T2
      TSE = TSE2
      XIN=XINT
 6051 PCA=PCA*DEGTOR
      SA=SA*DEGTOR
      ETNH=TN/COS(PCA)
      ETNS=ETNH/(COS(SA))**3.
      EFWH=FW
      EFWS=FW/COS(SA)
      RM=R-(FW/2.0)*SIN(PCA)
      RV=RM/COS(PCA)
      RS=RV/(COS(SA))**2.
      ALP=RM/R
      AV=ALP*A
      AS=AV
      ROV=RV+(AV*XIN)
      ROS=RS+(AS*XIN)
      BV=ALP*B
      BS=BV
      RRV=RV-(BV*XIN)
      RRS=RS-(BS*XIN)
      DM=ALP*D
      DV=DM
      DS=DV
      TIFRV=ROV-(DV*XIN)
      TIFRS=ROS-(DS*XIN)
      TM=ALP*T*COS(SA)
      TV=TM
      IF(N.EQ.2) TV = T
      TS=TV
      CV=(TIFRV-RRV)*XIN
      CS=(TIFRS-RRS)*XIN
      FRV=0.75*CV
      FRS=0.75*CS
      SA=SA/DEGTOR
      PCA=PCA/DEGTOR
      WRITE(6,10)
   10 FORMAT('1','TREGOLD APPROXIMATION FOR EQUIVALENT SPUR GEARS',/////)
      GO TO (32,20,9),N
    9 WRITE(6,1) I,TN,PCA,SA,FW,R,RM,ALP,A,B,D,T,TM
    1 FORMAT('0',    'SPIRAL BEVEL GEAR DESIGN DATA   GEAR NO.',4X,I2,//
     1       ' NUMBER OF TEETH - SPIRAL BEVEL       ',14X,F10.6/
```

```
     2' PITCH CONE ANGLE - SPIRAL BEVEL      ',14X,F10.6,4X,'DEGREES'/
     3' SPIRAL ANGLE OF SPIRAL BEVEL         ',14X,F10.6,4X,'DEGREES'/
     4' FACE WIDTH OF SPIRAL BEVEL           ',14X,F10.6,4X,'INCH  '/
     5' PITCH RADIUS OF SPIRAL BEVEL         ',14X,F10.6,4X,'INCH  '/
     6' MEAN PITCH RADIUS OF SPIRAL BEVEL    ',14X,F10.6,4X,'INCH  '/
     7' RATIO OF RADII OF SPIRAL BEVEL       ',14X,F10.6,4X,'INCH  '/
     8' ADDENDUM OF SPIRAL BEVEL AT LARGE END',13X,F10.6,4X,'INCH  '/
     9' DEDENDUM OF SPIRAL BEVEL AT LARGE END',13X,F10.6,4X,'INCH  '/
     1' WORKING DEPTH OF SPIRAL BEVEL AT LARGE END',8X,F10.6, 4X,'INCH'/
     2' CIR.TOOTH THICKNESS-SPIRAL BEVEL-LARGE END',8X,F10.6, 4X,'INCH'/
     3' MEAN CIR.TOOTH THICKNESS-SPIRAL BEVEL',13X,F10.6, 4X,'INCH')
   20 WRITE(6,21) I,ETNH,EFWH,RV,AV,ROV,HV,RRV,DV,TIFRV,TV,CV,FRV
   21 FORMAT(////' EQUIVALENT HELICAL GEAR       GEAR NO.',4X,I2,///
     1' EQUIVALENT NUMBER OF TEETH - HELICAL',14X,F10.6/
     2' EQUIVALENT FACE WIDTH - HELICAL      ',13X,F10.6, 4X,'INCH  '/
     3' EQUIVALENT PITCH RADIUS - HELICAL   ',15X,F10.6, 4X,'INCH  '/
     4' EQUIVALENT ADDENDUM - HELICAL ',20X,F10.6, 4X,'INCH  '/
     5' EQUIVALENT OUTER RADIUS - HELICAL ',16X,F10.6, 4X,'INCH  '/
     6' EQUIVALENT DEDENDUM - HELICAL ',20X,F10.6, 4X,'INCH  '/
     7' EQUIVALENT ROOT RADIUS - HELICAL',18X,F10.6, 4X,'INCH  '/
     8' EQUIVALENT WORKING DEPTH - HELICAL',16X,F10.6, 4X,'INCH  '/
     9' EQUIVALENT T.I.F. RADIUS - HELICAL',16X,F10.6, 4X,'INCH  '/
     1' EQUIVALENT CIRCULAR TOOTH THICKNESS-HELICAL',7X,F10.6,4X,'INCH'/
     2' EQUIVALENT RADIAL CLEARANCE-HELICAL',15X,F10.6, 4X,'INCH'/
     3' EQUIVALENT TOOTH FILLET RADIUS-HELICAL',12X,F10.6, 4X,'INCH')
   32 WRITE(6,31)  I,ETNS,EFWS,RS,AS,ROS,HS,RRS,DS,TIFRS,TS,CS,FRS
   31 FORMAT(////' EQUIVALENT SPUR GEAR      GEAR NO. ',4X,I2,///
     1' EQUIVALENT NUMBER OF TEETH - SPUR',17X,F10.6/
     2' EQUIVALENT FACE WIDTH - SPUR ',21X,F10.6, 4X,'INCH  '/
     3' EQUIVALENT PITCH RADIUS - SPUR ',19X,F10.6, 4X,'INCH  '/
     4' EQUIVALENT ADDENDUM - SPUR ',23X,F10.6, 4X,'INCH  '/
     5' EQUIVALENT OUTER RADIUS - SPUR',20X,F10.6, 4X,'INCH  '/
     6' EQUIVALENT DEDENDUM - SPUR',24X,F10.6, 4X,'INCH  '/
     7' EQUIVALENT ROOT RADIUS - SPUR',21X,F10.6, 4X,'INCH  '/
     8' EQUIVALENT WORKING DEPTH - SPUR ',18X,F10.6, 4X,'INCH  '/
     9' EQUIVALENT T.I.F. RADIUS - SPUR',19X,F10.6, 4X,'INCH  '/
     1' EQUIVALENT CIRCULAR TOOTH THICKNESS-SPUR',10X,F10.6, 4X,'INCH'/
     2' EQUIVALENT RADIAL CLEARANCE-SPUR',18X,F10.6, 4X,'INCH'/
     3' EQUIVALENT TOOTH FILLET RADIUS-SPUR',15X,F10.6, 4X,'INCH')
      EN(I)=ETNS
      RP(I)=RS
      RO(I)=ROS
      RT(I)=TIFRS
      ARP(I)=RRS
      AT(I)=TS
      F(I)=EFWS
      RF(I)=FRS
      RB(I)=RP(I)*COS(PAR)
      ARGR=RB(I)/RP(I)
      THETP=ARCOS(ARGR)/DEGTOR
      VPT(I)=TSE
      IF(IPROP.EQ.1) GO TO 40
      YE(I)=E
      POS(I)=ANU
   40 PE(I)=YE(I)/(2.0*(1.0+POS(I)) )
      IF(I.EQ.1) GO TO 6050
C END OF TREGOLD APPROXIMATION ROUTINE
C
C
      CL=RP(1) + RP(2) + DEV
      DPIT=EN(1)/(2*RP(1))
      IF(IHICP.EQ.1.AND.J.NE.4) GO TO 5550
 3001 GO TO (5555,3002,3003,3004),J
C
C
C R-33 OUTPUT ROUTINE
 3002 WRITE(6,4003)
 4003 FORMAT('1',5X,'THE FOLLOWING IS THE PUNCHED INPUT FOR R-33 GEARD',
     1//////)
      WRITE(6,4005) IDENT
 4005 FORMAT('0',10X,'CARD NUMBER 1'//,20A4)
      WRITE(7,4008) IDENT
```

```
 4008 FORMAT(20A4)
      WRITE(6,4009)MN,MMM
 4009 FORMAT('0',10X,'CARD NUMBER 2'//
     1              10X,'NUZ',10X,'INT',10X,'NMZ',10X,'MN',10X,'MMM'/
     2              ,11X,'1',12X,'1',12X,'1',11X,I1,12X,I2)
      WRITE(7,4012) MN,MMM
 4012 FORMAT(4X,'1',4X,'1',4X,'1',4X,I1,3X,I2)
      WRITE(6,4013)FN(1),FN(2),RP(1),THETP,RO(1),RO(2)
 4013 FORMAT('0',10X,'CARD NUMBER 3'//
     17X,'FN1',9X,'FN2',11X,'RP1',8X,'THETP',10X,'RO1',7X,'RO2 OR RI2'/
     2              ,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6)
      WRITE(7,4016) FN(1),FN(2),RP(1),THETP,RO(1),RO(2)
 4016 FORMAT(6E13.5)
      WRITE(6,4017)RT(1),RT(2),ARM(1),ARM(2),FI,AT(1)
 4017 FORMAT('0',10X,'CARD NUMBER 4'//
     1  , 8X,'RT1', 9X,'RT2',10X,'RM1',10X,'RM2',10X,'FI',11X,'T1'/
     2              ,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6)
      WRITE(7,4016) RT(1),RT(2),ARM(1),ARM(2),FI,AT(1)
      WRITE(6,4019) AT(2),F(1),F(2),RF(1),RF(2)
 4019 FORMAT('0',10X,'CARD NUMBER 5'//
     1              ,10X,'T2',10X,'F1',10X,'F2',10X,'RF1',10X,'RF2'/
     2              ,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6)
      WRITE(7,4022) AT(2),F(1),F(2),RF(1),RF(2)
 4022 FORMAT(5E13.5)
      WRITE(6,4023) YE(1),YE(2),GE(1),GE(2),POS(1),POS(2)
 4023 FORMAT('0',10X,'CARD NUMBER 6'//
     1              ,10X,'YE1',11X,'YE2',13X,'GE1',13X,'GE2',15X,'POS1', 9X,
     1'POS2'/
     3              ,3X,F13.5,3X,E13.5,3X,F13.5,3X,E13.5,3X,F13.6,3X,F10.6)
      WRITE(7,4016) YE(1),YE(2),GE(1),GE(2),POS(1),POS(2)
      NIT= 2.*(2.*FI+1.)
      DO 4025 I=1,NIT
      WRITE(7,4026)
 4026 FORMAT(80X)
 4025 CONTINUE
      WRITE(6,4027) NIT,WT,VPT(1),VPT(2)
 4027 FORMAT('0',10X,'CARDS NUMBER 7 HAS ',I3,' BLANK CARDS'//
     1,10X,'CARD NUMBER 8'//, 8X,'WT',9X,'VPT1',9X,'VPT2'/
     2              ,3X,F10.2,3X,F10.6,3X,F10.6)
      WRITE(7,4444)  WT,VPT(1),VPT(2)
 4444  FORMAT(3E13.5)
      GO TO 5555
C END  R-33 OUTPUT ROUTINE
C
C
C     BEGIN OUT PUT ROUTINE FOR R67 GGEAR PROGRAM
 3003 WRITE(6,4040)
 4040 FORMAT('1',5X,'THE FOLLOWING IS THE PUNCHED INPUT FOR R-67 GGEAR',
     1//////)
      WRITE(6,4005) IDENT
      WRITE(7,4008) IDENT
      WRITE(6,4043) MN,MMM,IPLT
 4043 FORMAT('0',10X,'CARD NUMBER 2'//
     1              ,10X,'NMC',10X,'INT',10X,'MN',10X,'MMM',10X,'IPLT',10X,
     2'IFOUR',10X,'ISPECT'/
     3              ,11X,'1',12X,'0',11X,I2,11X,I2,12X,I2,12X,'1',12X,'1')
      WRITE(7,4046)MN,MMM,IPLT
 4046 FORMAT(4X,'1',4X,'0',4X,I1,3X,I2,4X,I1,4X,'1',4X,'1')
      WRITE(6,4047)FN(1),FN(2),RB(1),RO(1),RO(2)
 4047 FORMAT('0',10X,'CARD NUMBER 3'//
     1  , 9X,'FN1', 9X,'FN2',10X,'RB1',10X,'RO1', 6X,'RO2 OR RI2'/
     2              ,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6)
      WRITE(7,4050) FN(1),FN(2),RB(1),RO(1),RO(2)
 4050 FORMAT(5E13.5)
      WRITE(6,4051) RT(1),RT(2),ARM(1),ARM(2),FI,AT(1)
 4051 FORMAT('0',10X,'CARD NUMBER 4'//
     1  , 9X,'RT1',10X,'RT2',10X,'RM1',10X,'RM2',10X,'FI',11X,'T1'/
     2              ,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6)
      WRITE(7,4054) RT(1),RT(2),ARM(1),ARM(2),FI,AT(1)
 4054 FORMAT(6E13.5)
      WRITE(6,4055) AT(2),F(1),F(2),RF(1),RF(2)
 4055 FORMAT('0',10X,'CARD NUMBER 5'//
```

```
     1            ,10X,'T2',10X,'F1',10X,'F2',10X,'RF1',10X,'RF2'/
     2            ,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6,3X,F10.6)
      WRITE(7,4050) AT(2),F(1),F(2),RF(1),RF(2)
      WRITE(6,4057) YE(1),YE(2),GE(1),GE(2),POS(1),POS(2)
 4057 FORMAT('0',10X,'CARD NUMBER 6'//
     1,10X,'YE1',12X,'YE2',12X,'GE1',13X,'GE2',11X,'POS1',10X,'POS2'/
     2            ,3X,E13.5,3X,E13.5,3X,E13.5,3X,E13.5,3X,F10.6,3X,F10.6)
      WRITE(7,4054) YE(1),YE(2),GE(1),GE(2),POS(1),POS(2)
      WRITE(6,4060)CL,VPT(1),VPT(2)
 4060 FORMAT('0',10X,'CARD NUMBER 7'//
     1  , 8X,'CL',10X,'VPT1',10X,'VPT2'/3X,F10.6,3X,F10.6,3X,F10.6)
      WRITE(7,4063) CL,VPT(1),VPT(2)
 4063 FORMAT(3E13.5)
       NIT=2.*FI+1
      DO 4064  I=1,NIT
      WRITE(7,4065)
 4065 FORMAT(80X)
 4064 CONTINUE
       DO 4066  I=1,NIT
      WT=TORK/RI
      WRITE(7,4067) WT
 4067 FORMAT(52X,E13.5)
 4066 CONTINUE
      WRITE(6,4068) NIT,NIT,NWS,WS
 4068 FORMAT('0',10X,'THE 8TH CARDS HAVE THE FIRST',2X,I2,2X,'CARDS BLAN
     1K'/          10X,'AND',2X,I2,2X,'CARDS WITH THE FIXED WT LOADING'//
     2 ,10X,'CARD NUMBER 9'//  10X,'NWS',10X,'WS'/
     3            ,11X,I2,5X,F10.3)
      WRITE(7,4073) NWS,WS
 4073 FORMAT(I5,E13.5)
 5555 GO TO 1999
C
C
C     BEGIN OUT PUT ROUTINE FOR R75 GGEAR/HCR PROGRAM
 3004 WRITE(6,4201)
 4201 FORMAT('1',5X,' THE FOLLOWING IS THE PUNCHED INPUT FOR R-75 GGEAR/
     1HCR'/////)
      WRITE(6,4005) IDENT
      WRITE(7,4008) IDENT
      WRITE(6,4209)MN,MMM,IPLT,IHICR
 4209 FORMAT('0',10X,'CARD NUMBER 2'//
     1            ,10X,'NMC',10X,'INT',10X,'MN',10X,'MMM',10X,'IPLT',10X,
     2'IFOUR', 8X,'ISPECT', 8X,'IHICR'/
     3 ,11X,'I',12X,'0',11X,I2,11X,I2,12X,I2,12X,'1',12X,'1',12X,I1)
      WRITE(7,4212) MN,MMM,IPLT,IHICR
 4212 FORMAT(4X,'1',4X,'0',3I5,4X,'1',4X,'1',I5)
      WRITE(6,4215) IRM,IPERC,IRF,IPROP,ICENT,NZJ,IDIAG
 4215 FORMAT('0',10X,'CARD NUMBER 2A'//10X,'IRM',9X,'IPERC',8X,'IRF',8X,
     1'IPROP',8X,'ICENT',8X'NZJ',8X,'IDIAG'/2X,7(11X,I1))
      WRITE(7,4216) IRM,IPERC,IRF,IPROP,ICENT,NZJ,IDIAG
 4216 FORMAT(7I5)
      WRITE(6,4217) FN(1),FN(2),DPIT,THETP,          RO(1),RO(2)
 4217 FORMAT('0',10X,'CARD NUMBER 3'//10X,'FN1',10X,'FN2',10X,'DPIT',8X,
     1'PANG',8X,'RO1',6X,'RO2 OR RI2'/6(3X,F10.6))
      WRITE(7,1360) FN(1),FN(2),DPIT,THETP,RO(1),RO(2)
      WRITE(6,4219)  ARM(1),ARM(2),FI,AT(1)
 4219 FORMAT('0',10X,'CARD NUMBER 4'//10X,'RM1',8X,'RM2',8X,'FI',10X,'T1
     1'/4(3X,F10.6))
      WRITE(7,1360) ARM(1),ARM(2),FI,AT(1)
      WRITE(6,4222)AT(2),F(1),F(2),RF(1),RF(2)
 4222 FORMAT('0',10X,'CARD NUMBER 5'//10X,'T2',10X,'F1',10X,'F2',10X,'RF
     11',10X,'RF2'/5(3X,F10.6))
      WRITE(7,1360) AT(2),F(1),F(2),RF(1),RF(2)
      IF(IPRCP.NE.1) GO TO 4230
      WRITE(6,4224) YE(1),YE(2),GE(1),GE(2),POS(1),POS(2)
```

```
 4224 FORMAT('0',10X,'CARD NUMBER 6 '//
     1,10X,'YE1',12X,'YE2',12X,'GE1',13X,'GE2',11X,'POS1',10X,'POS2'/
     2              ,3X,E13.5,3X,E13.5,3X,E13.5,3X,E13.5,3X,F10.6,3X,F10.6)
      WRITE(7,1360) YE(1),YE(2),GE(1),GE(2),POS(1),POS(2)
 4230 WRITE(6,4226)CL,VPT(1),VPT(2)
 4226 FORMAT('0',10X,'CARD NUMBER 7'//10X,'CL',10X,'VPT1',10X,'VPT2'
     1      /4(3X,F10.6))
      WRITE(7,1360) CL,VPT(1),VPT(2)
      NJ = 2.*FI+1
      MNJ=2*NJ
      IF(IHTCR.EQ.1) MNJ=4*NJ
      NZJTAB=NZJ+1
      GO TO(1350,1353,1353,1351,1350,1350),NZJTAB
 1350 READ(5,1360) (UJ1(I),I=1,MNJ)
 1360 FORMAT(6E13.5)
       READ(5,1360) (UJ2(I),I=1,MNJ)
      WRITE(6,1361) (UJ1(I),I=1,MNJ)
 1361 FORMAT('0',10X,'CARDS NUMBER 8A'/20X,'UJ1'/6E13.5)
      WRITE(6,1362) (UJ2(I),I=1,MNJ)
 1362 FORMAT('0',20X,'UJ2'/6E13.5)
      WRITE(7,1360) (UJ1(I),I=1,MNJ)
      WRITE(7,1360) (UJ2(I),I=1,MNJ)
 1351 GO TO(1352,1353,1353,1352,1353,1353),NZJTAB
 1352 READ(5,1360) (ZJ1(I),I=1,MNJ)
      READ(5,1360) (ZJ2(I),I=1,MNJ)
      WRITE(6,1363) (ZJ1(I),I=1,MNJ)
 1363 FORMAT('0',10X,'CARDS NUMBER 8B'/20X,'ZJ1'/3X,6E13.5)
      WRITE(6,1364) (ZJ2(I),I=1,MNJ)
 1364 FORMAT('0',20X,'ZJ2'/6E13.5)
      WRITE(7,1360) (ZJ1(I),I=1,MNJ)
      WRITE(7,1360) (ZJ2(I),I=1,MNJ)
 1353 IF(TORK.GT.0.01) GO TO 1355
      READ(5,1360) (TOR(I),I=1,NJ)
      GO TO 1356
 1355 DO 1357 I=1,NJ
      TOR(I)=TORK
 1357 CONTINUE
 1356 WRITE(6,1366) NTS
 1366 FORMAT('0',10X,'CARDS NUMBER 8C'//10X,'NTS'/10X,I1//10X,'TOR' )
      WRITE(6,1360) (TOR(I),I=1,NJ)
      WRITE(7,1370) NTS
 1370 FORMAT(I5)
      WRITE(7,1360) (TOR(I),I=1,NJ)
      IF(NZJTAB.NE.2.OR.NZJTAB.NE.5) GO TO 1354
      READ(5,1360) Z11,Z51,Z41,Z31,RHIGH1,RLOW1
      WRITE(6,1367) Z11,Z51,Z41,Z31,RHIGH1,RLOW1
 1367 FORMAT('0',10X,'CARDS NUMBER 8D'/20X,'Z1'/3X,6E13.5)
      READ(5,1360) Z12,Z52,Z42,Z32,RHIGH2,RLOW2
      WRITE(6,1368)Z12,Z52,Z42,Z32,RHIGH2,RLOW2
 1368 FORMAT(20X,'Z2'/3X,6E13.5)
      WRITE(7,1360) Z11,Z51,Z41,Z31,RHIGH1,RLOW1
      WRITE(7,1360) Z12,Z52,Z42,Z32,RHIGH2,RLOW2
 1354 CONTINUE
      WRITE(6,1385) NWS,WS
 1385 FORMAT('0',10X,'CARD NUMBER 9'//10X,'NWS',9X,'WS'/10X,I3,5X,F10.1)
      WRITE(7,1369) NWS,WS
 1369 FORMAT(I5,E13.5)
 1998 GO TO 1999
 5550 WRITE(6,5551) CR
 5551 FORMAT(////'  * *  THIS IS A HIGH CONTACT RATIO GEARSET'/15X,F10.4
     1/'    PROGRAM HAS BEEN RESET TO USE GGEAR/HCR * *')
      J=4
      GO TO 3001
 5556 STOP
      END
```

APPENDIX B

LISTING OF PROGRAM FOR PREDICTION OF GEAR MESH DYNAMICS FOR HIGH-CONTACT-RATIO SPUR GEARS (R-75)

```
FCRTI    IV C LEVEL  21                  MAIN                  DAT     75099         12/49/12                PAGE 0001
               C**** MEMBER NAME R75MAN                                                       00010000
               C                      MECHANICAL TECHNOLOGY INCORPORATED                      00020000
               C                           968 ALBANY-SHAKER ROAD                             00030000
               C                           LATHAM, NEW YORK 12110                             00040000
               C                               518-785-2211                                   00050000
               C                                                                              00060000
               C        COMPUTER PROGRAM FOR PREDICTION OF GEAR MESH EXCITATION SPECTRA       00070000
               C        SOURCE DECK                                                           00080000
               C        THIS PROGRAM REQUIRES SUBROUTINES GEARO, AJCOF, FOUR, CALCJ,          00090000
               C                                             PLT, SPECT                       00100000
               C                                                                              00110000
               C                                                                              
  C001                  DIMENSION CGG(3000),WTCC(3000),FJCC1(3000),AAAC(3000)                 00110100
  C002                  COMMON/YY/UJC1(25),UJD1(25),UJ2(25),UJC2(25),UJD2(25),ZJ1(50)         00120000
                       1,ZJC1(25),ZJD1(25),ZJ2(50),ZJC2(25),ZJD2(25),CJ1(50),QJC1(25),CJD100130000
                       2(25),QJ2(50),QJC2(25),                FJ1(50),UJ1(50),        CJC1(200140000
                       35),CJD1(25),         AJ1(50),AJ2(50),QJC(50),QJD(50),ZJC(50),ZJD(5000150000
                       4),VJ(50),EJ1(50),FJ2(50),AJC1(25),AJC2(25),FJC1(27),QJD2(25),ZM2( 00160000
                       550),UM2(50),WTC(27)                                                   00170000
  C003                  COMMON PJ(50),CJ(50),GJ(50),AJ(50),XJ(50),YJ(50),                     00180000
                       1A(1500),B(1500),C(3000)                                               00190000
  C004                  COMMON CC,ER,BA,FJ,CC,YM,RN,R,FNJ,EF,TAN,RB,GM,F,XM,MN,II,M           00200000
  C005                  COMMON QA(50),QB(50),QC(50),QA1(50),QB1(50),QC1(50),QA2(50),QB2(5000210000
                       1),QC2(50) ,IM,IF                                                      00220000
  C006                  COMMON WT(50),HEAD1(15),HEAD2(15)                                     00230000
  C007                  COMMON/GU/NMC,INT,MMM,IFLT,FN1,FN2,FB1,RO1,RO2,RT1,RT2,RM1,RM2,FI,00240000
                       1T1,T2,FIMF2,HF1,RF2,YE1,YE2,CE1,GE2,POS1,POS2,ANG                     00250000
  C008                  COMMON/INHI/ IHI(R,UJ(25,4,2),ZJ(25,4,2),WTT(25,4),VPT1,VPT2,CS,      00260000
                       1 SUMGJ(25,4),QJJ(25,4,2)                                              00270000
  C009                  COMMON/INVF/       IFCUR,ISPECT,IRM,IPERC,IRF,IPROP,ICENT,NZJ         00280000
  C010                  COMMON/CCEV/NR,NW,DPIT,FANG,BL,PT1,HT1,HT2,DEV                        00290000
                       1,TCRK(25),TCRC(25,8)                                                  00300000
  C011                  NR=5                                                                  00320000
  C012                  NW=6                                                                  00330000
  C013              156 READ(NR,100)(HEAD1(I),I=1,10),(HEAD2(I),I=1,10)                       00340000
  C014                  READ(NR,117) NMC,INT,MN,MMM,IFLT,IFCUR,ISPECT                         00350000
                       1 ,IHIC?                                                               00360000
  C015                  NEWIN=1                                                               00370000
  C016                  IF(NEWIN.EQ.0) GO TO 1580                                             00380000
  C017                  CALL BVIN(NNN,WTCC,FJCC1,CGG)                                         00390000
  C018                  GO TO 500                                                             00400000
  C019             1580 CONTINUE                                                              00410000
  C020                  READ(NR,112)FN1,FN2,RB1,RC1,RC2                                       00420000
  C021                  READ(NR,112)RT1,RT2,RM1,RM2,FI,T1                                     00430000
  C022                  READ(NR,112)T2,F1,F2,RF1,RF2                                          00440000
  C023                  READ(NR,112)YE1,YE2,CE1,GE2,POS1,POS2                                 00450000
  C024                  WRITE(NW,100)(HEAD1(I),I=1,10),(HEAD2(I),I=1,10)                      00460000
  C025                  WRITE(NW,101)                                                         00470000
  C026                  WRITE(NW,108)                                                         00480000
  C027                  WRITE(NW,105) NMC,INT,MN,MMM                                          00490000
  C028                  WRITE(NW,104)                                                         00500000
  C029                  WRITE(NW,102) F1,FN1,RB1,RC1,RT1,RM1                                  00510000
  C030                  IF(MN) 511,510,511                                                    00520000
```

```
  510 WRITE(NW,103)
      GO TO 512
  511 WRITE(NW,160)
  512 WRITE(NW,158)THETF,FN2,FC2,RT2,RM2
      WRITE(NW,105)
      WRITE(NW,102)T1,F1,RF1,YE1,GE1,POS1
      WRITE(NW,107)
      WRITE(NW,102)T2,F2,RF2,YE2,GE2,POS2
      WRITE(NW,164)
      ANG=0.
      IMC=1.
      N=2.*FI+1.
      NN=N+N
      IF(IMICR.NE.0) NN= 4*N
      IF(MN.EQ.0) T2=-T2
  150 WRITE(NW,1500)
      READ(NR,112)CL,VPT1,VPT2
      WRITE(NW,1000) IMC,CL
      WRITE(NW,110)
      IF(IMICR.NE.0) GO TO 400
      L=NN
      DO 209 I=1,NN
      READ(NR,112)ZJ1(I),UJ1(I),ZJ2(I),UJ2(I),WT(I)
      WRITE(NW,102)ZJ1(I),UJ1(I),ZJ2(I),UJ2(I)
      ZM2(L)=ZJ2(I)
      UM2(L)=UJ2(I)
      IF(L-1) 209,209,225
  225 L=L-1
  209 CONTINUE
      GO TO 410
  400 CONTINUE
      LL=5
      DO 210 I=1,4
      LL=LL-1
      L=N+1
      DO 210 J=1,N
      L=L-1
      READ(NR,112) ZJ(J,I,1),UJ(J,I,1),ZJ(L,LL,2),UJ(L,LL,2),WTT(J,I)
      WRITE(NW,102) ZJ(J,I,1),UJ(J,I,1),ZJ(L,LL,2),UJ(L,LL,2),WTT(J,I)
  210 CONTINUE
  410 CONTINUE
      CALL GEARC(CL,VPT1,VPT2)
      NNN=IMC*N
      DO 200 I=1,N
      JJ=NNN+I-N
      AANG(JJ)=AJC1(I)
      GGG(JJ)=EJ1(I)
      G(JJ)=GGG(JJ)
      WTCC(JJ)=WTC(I)
  200 FJCC1(JJ)=JJ
      IMC=IMC+1
      IF(IMC.LE.NMC) GO TO 150
```

```
FORT    IV G LEVEL  21                    MAIN              DA    = 75099         12/49/12

 0083                WRITE(NW,1500)                                                  01050000
 0084                WRITE(NW,2000)                                                  01060000
 0085                DO 2500 I=1,NNN                                                 01070000
 0086          2500  WRITE(NW,102)FJCC1(I),AANC(I),GGG(I)                            01080000
 0087           500  CONTINUE                                                        01090000
 0088                IF(IPLT.EQ.1) CALL PLT(FJCC1,GGG,WTCC,NNN)                      01100000
 0089                M=FI                                                            01110000
 0090                IF(IFOUR.NE.1) GO TO 3000                                       01120000
 0091                WRITE(NW,128)                                                   01130000
 0092                CALL FOUR                                                       01140000
 0093                LL=NYM+1                                                        01150000
 0094                DO 250 I=1,LL                                                   01160000
 0095                LP=I-1                                                          01170000
 0096                SMK=1.*LP/NMC                                                   01180000
 0097                RESLT=SQRT(A(I)**2+B(I)**2)                                     01190000
 0098           250  WRITE(NW,129)LP,A(I),B(I),RESLT,SMK                             01200000
 0099          3000  IF(ISPECT.NE.1) GO TO 3500                                      01210000
 0100                CALL SPECT                                                      01220000
 0101          3500  CONTINUE                                                        01230000
 0102                IF(INT)300,158,300                                              01240000
 0103           300  CALL EXIT                                                       01250000
 0104           100  FORMAT(20A4)                                                    01260000
 0105           101  FORMAT(72HC   PNC335*COMPUTATIONS OF GEAR TOOTH MESHING ERRORS      01270000
                    1  3-22-1966       )                                             01280000
 0106           102  FORMAT(7E13.5)                                                  01290000
 0107           103  FORMAT(12HC   PRE.ANGLE7X2HN224X3HRI210X3HRT21CX3HRM2)          01300000
 0108           104  FORMAT(/7X1HI111X2HN111X2HRB110X3HRO110X3HRT110X3HRM1)          01310000
 0109           105  FORMAT(/6X2HT111X2HF111X3HRF15X12HYOUNGS MOD-12X11HSHEAR MOD-12X1101320000
                    1HPOS.RATIO-1)                                                   01330000
 0110           107  FORMAT(/6X2HT211X2HF211)3HRF25X12HYOUNGS MOD-22X11HSHEAR MOD-22X1101340000
                    1HPOS.RATIO-2)                                                   01350000
 0111           108  FORMAT(50HC MESH CYCLES     INPUT,     DIFF.GEAR N.OF HARMONIC)  01360000
 0112           109  FORMAT(4X14,4(8X14))                                            01370000
 0113           110  FORMAT(60HCINPUT LISTING OF PROFILE ERROR AND SUPPLEMENTATY COMPLIC01380000
                    1ANCE/6X2HZ111X2HC111X2HZ211X2HU2)                               01390000
 0114           112  FORMAT(6E13.5)                                                  01400000
 0115           117  FORMAT(16I5)                                                    01410000
 0116           128  FORMAT(43HOCALCULATED FOURIER COEFFICIENTS FOR ERRORS//7H     19X01420000
                    15HARIK 13X4HB%1K14X1HC14X2HKM)                                  01430000
 0117           129  FORMAT(I7,3(4X  E14.7),F14.7)                                   01440000
 0118           158  FORMAT(2E13.5,13X,3E13.5)                                       01450000
 0119           160  FORMAT(12HC           7X2HN224X3HRO210X3HRT21CX3HRM2)           01460000
 0120           164  FORMAT(//16H CALCULATED DATA)                                   01470000
 0121          1000  FORMAT(16H MESH CYCLE NC.*,I3,20H,    CENTER DISTANCE*,E13.6)   01480000
 0122          1500  FORMAT(/,6X,20(1H*,3X),/)                                       01490000
 0123          2000  FORMAT(5X 3HJC19X4HAJC16X11HTANG. ERROR)                        01500000
 0124                STOP                                                            01510000
 0125                END                                                             01520000
```

```
FORTRAN IV G LEVEL  21                 AJCCH               DATE = 75099          12/49/12

 0001                 SUBROUTINE AJCCH                                                  00010000
                C**** MEMBER NAME R75AJCCH                                              00020000
 0002                 DIMENSION SD(50),CD(50),BIK(50),AIK(50),FLK(50) ,CDC(50)          00030000
 0003                 COMMON PJ(50),CJ(50),DJ(50),AJ(50),XJ(50),YJ(50),                  00040000
                     1A(1500),B(1500),G(3000)                                            00050000
 0004                 COMMON CC,ER,EA,FJ,CC,YM,NN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M       00060000
 0005                 COMMON CA(50),CB(50),CC(50),QA1(50),QB1(50),QC1(50),QA2(50),QB2(5000070000
                     1),CC2(50),IM,IF                                                    00080000
 0006                 DO 106 I=1,NN                                                      00090000
 0007                 CA(I)=0.0                                                          00100000
 0008                 CB(I)=0.0                                                          00110000
 0009                 CC(I)=0.0                                                          00120000
 0010                 CJ(I)=0.0                                                          00130000
 0011                 XJ(I)=0.0                                                          00140000
 0012             106 YJ(I)=0.0                                                          00150000
 0013                 IF(II-1) 405,405,403                                               00160000
 0014             405 MM=1.0                                                             00170000
 0015                 GO TO 404                                                          00180000
 0016             403 MM=MN                                                              00190000
 0017             404 K=1                                                                00200000
 0018                 DO 501 I=1,NN                                                      00210000
 0019                 AJ(I)=FJ*CC                                                        00220000
 0020                 PJ(I)=FJ                                                           00230000
 0021                 IF(II-1) 412,412,413                                               00240000
 0022             412 CALL CALCJ(AJ(I),ER,EA,XY)                                         00250000
 0023                 CJ(I)=XY                                                           00260000
 0024                 GO TO 414                                                          00270000
 0025             413 XY=CDC(I)                                                          00280000
 0026             414 IF(XY) 503,503,101                                                 00290000
 0027             101 IF(K-1) 102,102,103                                                00300000
 0028             102 IF=I                                                               00310000
 0029             103 AX=AJ(I)                                                           00320000
 0030                 DJ=CC+AX                                                           00330000
 0031                 SD(I)=SIN (DJ)                                                     00340000
 0032                 CD(I)=COS (DJ)                                                     00350000
 0033                 C1=ATAN (TAN*AX)                                                   00360000
 0034                 C3=COS (C1)                                                        00370000
 0035                 C4=C1-DJ                                                           00380000
 0036                 IF(MM) 402,401,402                                                 00390000
 0037             401 C4=-C4                                                             00400000
 0038             402 C5=SIN (C4)                                                        00410000
 0039                 C6=COS (C4)                                                        00420000
 0040                 C1=RB/C3                                                           00430000
 0041                 XJ(I)=C1*C6                                                        00440000
 0042                 YJ(I)=C1*C5                                                        00450000
 0043                 IF(K-1) 242,242,243                                                00460000
 0044             242 BIK(I)=(YJ(I)**3+YM**3)/3.0*F                                      00470000
 0045                 AIK(I)=(YJ(I)+YM)*F                                                00480000
 0046                 FLK(I)=XJ(I)-XM                                                    00490000
 0047                 GO TO 504                                                          00500000
 0048             243 BIK(I)=(YJ(I)**3+YJ(I-1)**3)/3.0*F                                 00510000
 0049                 AIK(I)=(YJ(I)+YJ(I-1))*F                                           00520000
```

```
FORTRAN IV G LEVEL  21                    AJCEH              DATE = 75099         12/49/12

            FLK(I)=XJ(I)-XJ(I-1)
      504   K=K+1
      503   IF(I-NN) 502,501,501
      502   IF(MM) 506,505,506
      505   FJ=FJ-1.
            GO TO 501
      506   FJ=FJ+1.0
      501   CONTINUE
            IF(II-1) 301,301,302
      301   IF(MM) 601,302,601
      601   LL=NN
            DO 303 I=1,NN
            CCC(I)=CJ(LL)
            IF(LL-1) 303,303,305
      305   LL=LL-1
      303   CONTINUE
      302   IP=K+IM-2
      410   DO 246 L=IM,IP
            E1=0.0
            E2=0.0
            E3=0.0
            E4=0.0
            EE=XJ(L)-XM
            IM=L
            IF(MM) 202,203,202
      203   EE=-EE
      202   DO 245 I=IM,IM
            C1       =XJ(L)-XJ(I)
            C5=FLK(I)
            IF(MM) 207,208,207
      208   C1=-C1
            C5=-C5
      207   C2=C5/BIK(I)
            C3=(C5+3.0*C1)*C5+3.0*C1*C1
            E4=E4+C2
            C4=2.0*C1+C5
            E1=E1+C3*C2
            E2=E2+C4*C2
      245   E3=E3+C5/AIK(I)
            IF(MM) 206,205,206
      205   IM=IM-1
      206   C1=CD(L)*CC(L)
            C2=SD(L)*YJ(L)
            CA(L) =C1/2.0*E1/EP+(C2*E4-CC(L)*E2)/EP*C2
            CB(L) =C1/(M*E3*1.2
            C1=(EE*CC(L)-YJ(L)*SC(L))/YM
            CC(L)=C1/EF*1.327/F*C1
      246   CJ(L)=QA(L)+CB(L)+CC(L)
            RETURN
            END
```

```
FORTRAN IV G LEVEL  21                BVIN                DATE = 75099          12/49/12

 0001          SUBROUTINE BVIN(NNN,WTCC,FJCC1,GGG)                                    00010000
        C****  MEMBER NAME R75BVIN                                                    00020000
 0002          DIMENSION GGG(3000),WTCC(3000),FJCC1(3000),AANG(3000)                  00030000
 0003          DIMENSION HZJ(25,4),HUJ(25,4),ROLL(3,2)                                00040000
 0004          COMMON PJ(50),CJ(50),CJ(50),AJ(50),XJ(50),YJ(50),                      00050000
              1A(1500),B(1500),G(3000)                                                00060000
 0005          COMMON CC,BP,BA,FJ,CC,YM,NN,N,FNJ,EF,TTAN,RB,CM,F,XM,MN,II,M           00070000
 0006          COMMON CA(50),CB(50),CC(50),CA1(50),QB1(50),QC1(50),QA2(50),QB2(5000080000
              1),CC2(50) ,IH,IF                                                       00090000
 0007          COMMON WT(50),HEAD1(15),HEAD2(15)                                      00100000
 0008          COMMON/CL/NNC,INT,MMM,IFLT,FN1,FN2,RB1,RO1,RO2,RT1,RT2,RM1,RM2,FI,C0110000
              1T1,T2,F1,F2,RF1,RF2,YE1,YE2,CE1,GE2,POS1,PCS2,ANG                      00120000
 0009          COMMON/INFI/IHICF,CJ(25,4,2),ZJ(25,4,2),HTT(25,4),VPT1,VPT2,CS,        00130000
              1 SUMCJ(25,4),QJJ(25,4,2)                                               00140000
 0010          COMMON/INVR/      IFCLR,ISFECT,IRM,IPERC,IRF,IPROP,ICENT,NZJ           00150000
 0011          COMMON/CCBV/NR,NW,DPIT,FANG,BL,PT1,HT1,HT2,DEV                         00160000
              1,TCRK(25),TCFQ(25,E)                                                   00170000
 0012          COMMON/TAB/ ALPH(5,2),Z(5,2),RLOW(2),RHIGH(2)                          00180000
 0013          COMMON/DIAG/ IDIAG                                                     00190000
 0014          COMMON/YY/     UJC1(25),UJD1(25),UJ2(25),UJC2(25),UJD2(25),ZJ1(50)C0200000
              1,ZJC1(25),ZJD1(25),ZJ2(50),ZJC2(25),ZJD2(25),CJ1(50),QJC1(25),QJD100210000
              2(25),QJ2(50),QJC2(25),               FJ1(50),UJ1(50),       CJC1(200220000
              35),CJD1(25),          AJ1(50),AJ2(50),QJC(50),QJD(50),ZJC(50),ZJD(50C0230000
              4),VJ(50),EJ1(56),FJ2(50),AJC1(25),AJC2(25),FJC1(27),QJD2(25),ZM2(      00240000
              550),UM2(50),WTC(27)                                                    00250000
 0015          DATA PI/3.1415926E/                                                    00270000
 0016          YE1=3.E7                                                               00270300
 0017          YE2=3.E7                                                               00270400
 0018          GE1=1.15E7                                                             00270500
 0019          GE2=1.15E7                                                             00270600
 0020          POS1=.3                                                                00270700
 0021          POS2=.3                                                                00270800
 0022          READ(NR,117) IRM,IPERC,IRF,IPROP,ICENT,NZJ                             00280000
              1 ,IDIAG                                                                00290000
 0023          READ(NR,112) FN1,FN2,DPIT,PANG,RO1,RO2                                 00300000
 0024          IF(MN.EQ.0) RI2=RO2                                                    00310000
 0025          RB1=FN1*COS(PANG*PI/180.)/2./DPIT                                      00320000
 0026          RB2=RB1*FN2/FN1                                                        00330000
 0027          THETP=0.0                                                              00340000
 0028          READ(NR,112) RM1,RM2,FI,T1                                             00350000
 0029          IF(IRM.EQ.0) GO TO 10                                                  00360000
 0030          READ(NR,112) HT1,HT2                                                   00370000
 0031          RM1=RO1-HT1                                                            00380000
 0032          IF(MN.EQ.1) RM2=RO2-HT2                                                00390000
 0033          IF(MN.EQ.0) RM2=RI2+HT2                                                00400000
 0034       10 CONTINUE                                                               00410000
 0035          READ(NR,112) T2,F1,F2,RF1,RF2                                          00420000
 0036          IF(IPERC.EQ.0) GO TO 20                                                00430000
 0037          READ(NR,112) BL,FT1                                                    00440000
 0038          PC=PI/DPIT                                                             00450000
 0039          T1=(PC-BL)*FT1                                                         00460000
 0040          T2=(PC-BL)*(1.-FT1)                                                    00470000
```

```
   20 CONTINUE
      IF(IRF.NE.0) GO TO 25
      RT1=RB1
      RT2=RB2
      IF(MN.EQ.0) RT2=RM2
      SARG=(RF1+RM1)**2-RB1**2
      IF(SARG.LT.0.) GO TO 22
      GK=-RF1+SQRT(SARG)
      RT1=SQRT(RB1**2+GK**2)
   22 CONTINUE
      IF(MN.EQ.1) SARG=(RF2+RM2)**2-RB2**2
      IF(MN.EQ.0) SARG=(RM2-RF2)**2-RB2**2
      IF(SARG.LT.0.) GO TO 30
      IF(MN.EQ.1) GK=-RF2+SQRT(SARG)
      IF(MN.EQ.0) GK=RF2+SQRT(SARG)
      RT2=SQRT(RB2**2+GK**2)
      GO TO 30
   25 CONTINUE
      RT1=RM1
      RT2=RM2
      GK=SQRT(RT1**2-RB1**2)
      RF1=(RT1**2-RM1**2)/2./(RM1-GK)
      GK=SQRT(RT2**2-RB2**2)
      RF2=(RT2**2-RM2**2)/2./(RM2-GK)
      IF(MN.EQ.0) RF2=-RF2
   30 CONTINUE
      IF( IPRCP.EQ.1) READ(NR,112)YE1,YE2,GE1,GE2,POS1,POS2
      WRITE(NW,100)(HEAD1(I),I=1,10),(HEAD2(I),I=1,10)
      WRITE(NW,101)
      WRITE(NW,108)
      WRITE(NW,109)NMC,INT,FN,FFM,IFLT,IFCUR,ISPECT,IHICR,
     1IRM,IPERC,IRF,IPRCP,ICENT,NZJ
      WRITE(NW,104)
      WRITE(NW,102) FI,FN1,CRIT,RC1,RT1,RM1
      IF(IRM.NE.0) WRITE(NW,106) HT1,HT2
      IF(MN.EQ.0) WRITE(NW,103)
      IF(MN.NE.0) WRITE(NW,160)
      WRITE(NW,161) FN2,FANG,RC2,RT2,RM2
      IF(IPERC.NE.0) WRITE(NW,107) PT1,BL
      WRITE(NW,105)
      WRITE(NW,102) T1,F1,RF1,YE1,GE1,POS1
      WRITE(NW,107)
      WRITE(NW,102) T2,F2,RF2,YE2,GE2,POS2
      WRITE(NW,164)
      ANG= 0.
      INC= 1
      N = 2.*FI+1.
      NN = N + N
      IF(IHICR.NE.0) NN = 4*N
      IF(MN.EQ.0) T2= -T2
  150 CONTINUE
      WRITE(NW,1500)
```

```
C092          READ(NR,112) CL,VPT1,VPT2,DEV                          C1000000
C093          IF(ICENT.NE.C) CL=(FN1+FN2)/2./DPIT                     C1010000
C094          IF(ICENT.EQ.1) WRITE(NW,1C10) DEV                       C1020000
C095          IF(ICENT.EQ.1) CL=CL+DEV                                01030000
C096          FP1=CL*FN1/(FN1+FN2)                                    C1040000
C097          IF(NN.EQ.C) FP1=CL*FN1/(FN2-FN1)                        C1050000
C098          FP2=RP1*FN2/FN1                                         C1060000
C099          WRITE(NW,1COC) IHC,CL,VPT1,VPT2                         C1070000
C10C          IF(IHICH.NE.C) GC TC 4CC                                C1080000
      C       SET  ALL   ZJS  AND  UJS  TO  0                         C1090000
C101          DC 205 I=1,NN                                           C1100000
C102          ZJ1(I) = 0.0                                            C1110000
C103          UJ1 (I) = C.C                                           C1120000
C104          ZJ2 (I) = C.C                                           C1130000
C105          UJ2 (I) = C.C                                           C1140000
C106          TCRK (I) = C.0                                          C1150000
C107      205 CCNTINUE                                                C1160000
C108          IF(NZJ.EQ.C. OR .NZJ.EQ.3)                              C1170000
             1READ(NR,112) (ZJ1(I),I=1,NN)                            C1180000
C109          IF(NZJ.EQ.C. CR.NZJ.EQ.3)                               C1190000
             1READ(NR,112 )  (ZJ2(I),I=1,NN)                          C1200000
C11C          IF(NZJ.EQ.C. CR. NZJ. GT.3)                             C1210000
             1READ (NR,112)  (UJ1(I),I=1,NN)                          C1220000
C111          IF(NZJ.EQ.C.CR. NZJ.GT.3)                               C1230000
             1READ(NR,112) (UJ2(I),I=1,NN)                            C1240000
C112          READ (NR,117)  NTS                                      C1250000
C113          READ (NR,112)  (TCRK(I),I=1,NTS)                        C1260000
C114          IF(NZJ.GT.3) GC TC 3500                                 C1270000
C115          IF(NZJ.EQ.1.CR.NZJ.EQ.2) GC TC 2090                     C1280000
C116     205C CCNTINUE                                                C1290000
C117          L = NN                                                  C1300000
C118          IF(NZJ.EQ.C.CR.NZJ.EQ.3)                                C1310000
             1WRITE(NW,11C)                                           C1320000
C119          DC  209  I =1,NN                                        C1330000
C12C          IF(NZJ.EQ.C.CR.NZJ.EQ.3)                                C1340000
             1WRITE(NW,1C2) ZJ1(I),UJ1(I),ZJ2(I),UJ2(I)               C1350000
C121          ZM2(L)=ZJ2(I)                                           C1360000
C122          UM2 (L) = UJ2(I)                                        C1370000
C123          IF(L.GT.1)  L=L-1                                       C1380000
C124      209 CCNTINUE                                                C1390000
C125     209C CCNTINUE                                                C1400000
C126          DIV = RP1                                               C1410000
C127          IF (NTS.EQ.2) DIV = RP2                                 C1420000
C128          DC  320  I=1,NN                                         C1430000
C129          WT (I)= TCRK(I) / DIV                                   C1440000
C13C      32C CCNTINUE                                                C1450000
C131          WRITE(NW,1C9C)                                          C1460000
C132          IF (NTS.EC.1) WRITE(NW,1C4C)                            C1470000
C133          IF (NTS.EC.2) WRITE(NW,1C50)                            C1480000
C134          WRITE (NW,111C) (TCRK(I) , I=N,NN)                      C1490000
C135          WRITE(NW,11CC)                                          01500000
C136          WRITE(NW,111C)  (WT(I),I = N,NN)                        C1510000
```

FORTRAN IV G LEVEL 21 EVIN DATE = 75099 12/49/12

```
      IF(NZJ.NE.1.OR.NZJ.NE.4) GO TO 410
   35 CONTINUE
      DO 40 J=1,2
      Z(2,J) = 0.0
      READ(NR,112) Z(1,J),Z(5,J),Z(4,J),Z(3,J),RHIGH(J),RLOW(J)
   40 CONTINUE
      SRCCT=  SQRT((RC1/ RB1)**2 - 1. )
      PHI=PANG*PI/180.
      TANP=TAN(PHI)
      ALPH(1,1) = SRCCT - TANP
      SRCCT = SQRT((RC2/RB2)**2 - 1. )
      ALPH(1,2) = SRCCT - TANP
      ALPH(2,1) = 0.0
      ALPH(2,2) = 0.0
      ALPH(3,1) = -FN2/FN1 *( ALPH(1,2) )
      ALPH(3,2) = -FN1/FN2 *( ALPH(1,1) )
      IF(MN.NE.0) GO TO 45
      ALPH(3,2)=-ALPH(3,2)
      ALPH(3,1)=-RB2/RB1*(TANP-SRCCT)
      ALPH(1,2)=FN1/FN2*ALPH(3,1)
   45 CONTINUE
      DO  50  J=1,2
      ALPH(4,J)=RLOW(J)*PI/180.  -TANP
      ALPH(5,J)=RHIGH(J)*PI/180.-TANP
      DO 46 I=1,3
      ROLL(I,J)=(ALPH(I,J)+TANP)*180./PI
   46 CONTINUE
      WRITE(NW,1020)
      IF(J.EQ.1)  WRITE(NW,1024)
      IF(J.EQ.2) WRITE(NW,1025)
      IF(J.EQ.1.OR.MN.EQ.1) WRITE(NW,1021)
      IF(J.EQ.2.AND.MN.EQ.0) WRITE(NW,1022)
      WRITE(NW,1023) Z(1,J),Z(5,J),Z(2,J),Z(4,J),Z(3,J),
     1 ROLL(1,J),RHIGH(J),ROLL(2,J),RLOW(J),ROLL(3,J)
   50 CONTINUE
      IF(IDIAG.EQ.1) WRITE(6,5010) ALPH,Z
 5010 FORMAT(5X,10E12.4)
C     ARRANGE ALPH IN   ASCENDING  ORDER
      DO  80  J = 1,2
      DO  80  I = 2,5
      I = L
   60 CONTINUE
      IF(ALPH(I-1,J).LT.ALPH(I,J)) GO TO 70
      SWITCH=  Z (I-1,J)
      Z(I-1,J) = Z(I,J)
      Z(I,J  )  = SWITCH
      SWITCH=  ALPH(I-1,J)
      ALPH(I-1,J) = ALPH(I,J)
      ALPH(I,J)=SWITCH
   70 CONTINUE
      I=I-1
      IF(I.GE.2) GO TO 60
```

```
FORTRAN IV G LEVEL  21                BVIN              DATE = 75099        12/49/12

 0187          80 CONTINUE                                                   02040000
 0188             IF(IDIAG.EQ.1) WRITE(6,2000) ALPH,Z                        02050000
 0189             GO TO 410                                                  02060000
 0190        3500 CONTINUE                                                   02070000
            C     WRITE   ALL   U  VALUES                                    02080000
 0191             WRITE(NW,1120)                                             02090000
 0192             WRITE(NW,1040)                                             02100000
 0193             WRITE(NW,102) (UJ1(I),I=1,NN)                              02110000
 0194             WRITE(NW,1120)                                             02120000
 0195             WRITE(NW,1050)                                             02130000
 0196             WRITE(NW,102) (UJ2(I),I=1,NN)                              02140000
 0197             GO TO 2050                                                 02150000
 0198        3510 CONTINUE                                                   02160000
 0199             WRITE(NW,1120)                                             02170000
 0200                   WRITE(NW,1040)                                       02180000
 0201             WRITE(NW,102) ((UJ(J,I,1),J=1,N),I=1,4)                    02190000
 0202             WRITE(NW,1120)                                             02200000
 0203             WRITE(NW,1050)                                             02210000
 0204             WRITE(NW,102) ((HUJ(J,1),J=1,N),I=1,4)                     02220000
 0205             GO TO 4450                                                 02230000
 0206         400 CONTINUE                                                   02240000
            C     SET   ALL   U, Z  AND  TORK  TO 0                          02250000
 0207             DO 445 I=1,25                                              02260000
 0208             DO 445 J=1,4                                               02270000
 0209             TORQ(I,J)=0.                                               02271000
 0210             DO 445 K=1,2                                               02280000
 0211             UJ(I,J,K)=0.0                                              02290000
 0212             ZJ(I,J,K)=0.0                                              02300000
 0213         445 CONTINUE                                                   02320000
 0214             IF( NZJ.EQ.0.OR.NZJ.EQ.3 )                                 02330000
                 1 READ(NR,112) ((ZJ(J,I,1),J=1,N),I=1,4)                    02340000
 0215             IF(NZJ.EQ.0.OR.NZJ.EQ.3)                                   02350000
                 1 READ(NR,112) ((HZJ(J,I),J=1,N),I=1,4)                     02360000
 0216             IF(NZJ.EQ.0.OR.NZJ.GT.3)                                   02370000
                 1READ(NR,112)  ((UJ(J,I,1),J=1,N),I=1,4)                    02380000
 0217             IF(NZJ.EQ.0.OR.NZJ.GT.3)                                   02390000
                 1READ(NR,112)((HUJ(J,I),J=1,N),I=1,4)                       02400000
 0218             READ(NR,117) NTS                                           02410000
 0219             READ(NR,112)  (TORQ(J,3),J=1,N)                            02420000
 0220             IF(NZJ.GT.3) GO TO 3510                                    02430000
 0221             IF(NZJ.EQ.1.OR.NZJ.EQ.2) GO TO 2100                        02440000
 0222        4450 CONTINUE                                                   02450000
 0223             LL = 5                                                     02460000
 0224             IF(NZJ.EQ.0.OR.NZJ.EQ.3)                                   02470000
                 1WRITE(NW,110)                                              02480000
 0225             DO  210  I=1,4                                             02490000
 0226             L = N+1                                                    02500000
 0227             LL = LL-1                                                  02510000
 0228             DO  210  J=1,N                                             02520000
 0229             L = L-1                                                    02530000
 0230             IF(NZJ.EQ.0.OR.NZJ.EQ.3)                                   02540000
                 1WRITE(NW,102) ZJ(J,I,1),UJ(J,I,1),HZJ(J,I),HUJ(J,I)        02550000
```

```
      ZJ(L,LL,2)=HZJ(L,I)
      UJ(L,LL,2)=HUJ(L,I)
  210 CONTINUE
 2100 CONTINUE
      DIV=RP1
      IF(NTS.EQ.2) DIV=RP2
      DO 460 J=1,N
      WTT(J,3)=TORQ(J,3)/DIV
  460 CONTINUE
      WRITE(NW,1050)
      IF(NTS.EQ.1) WRITE(NW,1040)
      IF(NTS.EQ.2) WRITE(NW,1050)
      WRITE(NW,1110) (TORQ(J,3),J=1,N)
      WRITE(NW,1100)
      WRITE(NW,1110) (WTT(J,3),J=1,N)
      IF(NZJ.EQ.1.OR.NZJ.EQ.4) GO TO 35
  410 CONTINUE
      IF(IDIAG.GT.0) WRITE(6,5000) ZJ
 5000 FORMAT(3H ZJ,(10E12.5))
      CALL GEARC(CL,VPT1,VPT2)
      NNN=IMC*N
      DO 200 I=1,N
      JJ=NNN+I-N
      AANC(JJ)=AJC1(I)
      GGC(JJ)=SJ1(I)
      G(JJ)=GGC(JJ)
      WTC(JJ)=WTC(I)
  200 EJCC1(JJ)=JJ
      IMC=IMC+1
      IF(IMC.LE.NMC) GO TO 150
      WRITE(NW,1000)
      WRITE(NW,2000)
      DO 2500 I=1,NNN
 2500 WRITE(NW,102) EJCC1(I),AANC(I),GGG(I)
      RETURN
  100 FORMAT(20A4)
  101 FORMAT(72HC  RNC335*CORRELATIONS OF GEAR TOOTH MESHING ERRORS
     1  3-22-1968                    )
  102 FORMAT(5E13.5)
  103 FORMAT(/ 15X,2HN2, 5X,14HCLT. PRESS ANG,3X,3HRI2,10X,3HRT2,10X,
     13HRR2)
  104 FORMAT(/ 7X ,1HI,11X,2HN1,6X,10HDIA. PITCH,6X,3HRO1,10X,3HRT1,
     1 10X,3HSR1)
  105 FORMAT(/6X,2HT1,11X,2HF1,11X,3HRF1,5X,12HYOUNGS MOD-1,2X,
     1 11HSHEAR MOD-1,2X,11HPOS.RATIC-1)
  107 FORMAT(/6X,2HT2,11X,2HF2,11X,3HRF2,5X,12HYOUNGS MOD-2,2X,
     1 11HSHEAR MOD-2,2X,11HPOS.RATIC-2)
  108 FORMAT(90HC  NMC  INT  NN   MMM  IPLT IFOUR ISPECT IHICR  IRM IPER
     1C  IRF IFREP ICENT  NZJ                 )
  109 FORMAT(I6,I5,I4,I6,I5,2I6,I7,I4,2I5,3I6)
  110 FORMAT(60HCINPUT LISTING OF PROFILE ERROR AND SUPPLEMENTARY COMPLI
     1ANCE/6X,2HZ1,11X,2HU1,11X,2HZ2,11X,2HU2)
```

```
C276      112 FORMAT(6E13.5)                                                  03080000
C277      117 FORMAT(16I5)                                                    03090000
C278      158 FORMAT(2E13.5,13X,3E13.5)                                       03100000
C279      160 FORMAT(/ 15X,2HN2,5X,14HCUT. PRESS ANG,3X,3HRC2,10X,3HRT2,10X,  03110000
             1 3HRR2)                                                         03120000
C280      161 FORMAT(13X,5E13.5)                                              03130000
C281      164 FORMAT(//16H CALCULATED DATA    )                               03140000
C282     1000 FORMAT(16H MESH CYCLE NO.*,I3,20H,    CENTER DISTANCE*,E13.6    03150000
             1,4H IN./                                                        03160000
             143H VPT1, TOOTH SPACING ERROR %DRIVING GEAR< *    ,E13.6,4H IN. /03170000
             243H VPT2, TOOTH SPACING ERROR %DRIVEN  GEAR< *    ,E13.6,4H IN. )03180000
C283     1010 FORMAT(/    43H DEVIATION FROM STANDARD CENTER DISTANCE *     ,  03190000
             1 E13.6,4H IN.)                                                  03200000
C284     1020 FORMAT(///20X,70HDEVIATION OF POINT ON ACTUAL TOOTH PROFILE FROM T03210000
             1TRUE INVOLUTE                    )                              03220000
C285     1021 FORMAT(20X,75HAT OUTER DIAM.   HIGH POINT      PITCH POINT    LOW 03230000
             1POINT   TRUE INVOLUTE     / )                                   03240000
C286     1022 FORMAT(20X,75HAT INNER DIAM.   HIGH POINT      PITCH POINT    LOW 03250000
             1POINT   TRUE INVOLUTE     / )                                   03260000
C287     1023 FORMAT(6X,1HZ,14X,5(1X,E13.5)/6X,10HROLL ANGLE,5X,5(1X,E13.5))  03270000
C288     1024 FORMAT(40X,12HDRIVING GEAR  )                                   03280000
C289     1025 FORMAT(40X,12HDRIVEN GEAR          )                            03290000
C290     1030 FORMAT(18X,5E13.5,6X,2E13.5)                                    03300000
C291     1040 FORMAT(1H0,5X,12HDRIVING GEAR    )                              03310000
C292     1050 FORMAT(1H0,5X,12HDRIVEN  GEAR    )                              03320000
C293     1060 FORMAT(/5X,30HTOOTH DEPTH %DRIVING GEAR<    *,E13.5/            03330000
             1          /5X,30HTOOTH DEPTH %DRIVEN  GEAR<    *,E13.5)         03340000
C294     1070 FORMAT(/5X,75HPER CENT DIFFERENCE BETWEEN ACTUAL CIRCULAR PITCH AN03350000
             1D WORKING BACKLASH  *     ,E13.5    /                           03360000
             1 5X,10HBACKLASH *   ,E13.5)                                     03370000
C295     1080 FORMAT(5E13.5,I5)                                               03380000
C296     1090 FORMAT(/20X,50HTRANSMITTED TORQUE %IN-LB<                       )03390000
C297     1100 FORMAT(/20X,80HTOTAL LOAD TANGENT TO PITCH CIRCLE, TRANSMITTED BY 03400000
             1GEAR TEETH %LB<                                                 )03410000
C298     1110 FORMAT(2X,10E12.4)                                              03420000
C299     1120 FORMAT(/20X,40HTOOTH SUPPORT COMPLIANCE  %IN/LB<            )   03430000
C300     1500 FORMAT(/6X,20(1H*,2X)/)                                         03440000
C301     2000 FORMAT(5X,3HJC1,5X,4HAJC1,6X,11HTANG. ERROR    )                03450000
C302          END                                                             03460000
```

```
FORTRAN IV G LEVEL  21                 CALCJ             DATE - 75099          12/49/12

 0001              SUBROUTINE CALCJ(C1,C2,C3,C4)                                     00010000
            C**** MEMBER NAME R75CALCJ                                               00020000
 0002              IF(C3) 240,231,241                                                00030000
 0003          241 IF(C1) 230,218,218                                                00050000
 0004          230 IF(ABS (C1)-C3)  220,220,219                                      00060000
 0005          231 IF(C1) 219,218,218                                                00070000
 0006          240 IF(C1) 219,219,242                                                00080000
 0007          242 C3=-C3                                                            00090000
 0008              IF(C1-C3) 219,218,218                                             00100000
 0009          218 IF(C2-C1) 219,220,220                                             00110000
 0010          220 C4=1.0                                                            00120000
 0011              GO TO 221                                                         00130000
 0012          219 C4=0.0                                                            00140000
 0013          221 RETURN                                                            00150000
 0014              END                                                               00160000
```

```
FORTRAN IV G LEVEL  21                 FOUR                DATE  75099       12/49/12

 0001            SUBROUTINE FOUR                                                   00010000
           C**** MEMBER NAME R75FOUR                                               00020000
           C     REQUIRES 2*N+1 POINTS  F(I)                                       00030000
           C     POINTS CORRESPOND TO THETA=2*PI/(2*N+1),...,2*PI                  00040000
           C     OUTPUT  A(I),B(I)  REFER TO COSINE AND SINE OF (I-1)*THETA        00050000
 0002            COMMON PJ(50),CJ(50),QJ(50),AJ(50),XJ(50),YJ(50),                 00060000
                1A(1500),B(1500),G(3000)                                           00070000
 0003            COMMON DD,BR,BA,FJ,CC,YM,NN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M       00080000
 0004            COMMON/GL/NMC,INT,MMM,IFLT,FN1,FN2,RB1,RO1,RO2,RT1,RT2,RM1,RM2,FI,00090000
                1T1,T2,F1,F2,RF1,RF2,YE1,YE2,GE1,GE2,POS1,POS2,ANG                 00100000
 0005            C=1.0                                                             00110000
 0006            S=0.0                                                             00120000
 0007            M=IFIX(FI)*NMC+NMC/2                                              00130000
 0008            N1=M+1                                                            00140000
 0009            NA=N*NMC                                                          00150000
 0010            N2=NA-1                                                           00160000
 0011            T1=2./NA                                                          00170000
 0012            T2=T1*3.1415927                                                   00180000
 0013            C1=COS (T2)                                                       00190000
 0014            S1=SIN (T2)                                                       00200000
 0015            DO 7 IP=1,N1                                                      00210000
 0016            U1=0.0                                                            00220000
 0017            U2=0.0                                                            00230000
 0018            DO 3 I=1,N2                                                       00240000
 0019            J=N2-I+2                                                          00250000
 0020            U3=G(J)+2.C*C*U1-U2                                               00260000
 0021            U2=U1                                                             00270000
 0022          3 U1=U3                                                             00280000
 0023            A(IP)=T1*(G(1)+C*U1-U2)                                           00290000
 0024            B(IP)=T1*S*U1                                                     00300000
 0025            AA=A(IP)*C-B(IP)*S                                                00310000
 0026            BB=A(IP)*S+B(IP)*C                                                00320000
 0027            A(IP)=AA                                                          00330000
 0028            B(IP)=BB                                                          00340000
 0029            Q=C1*C-S1*S                                                       00350000
 0030            S=C1*S+S1*C                                                       00360000
 0031          7 C=Q                                                               00370000
 0032            RETURN                                                            00380000
 0033            END                                                               00390000
```

```
0001             SUBROUTINE GEARC(CL,LPT1,YE12)                                        00010000
           C**** MEMBER NAME R75GEARC                                                  00020000
0002             COMMON/YY/UJC1(25),UJD1(25),UJ2(25),UJC2(25),UJD2(25),ZJ1(50)         00030000
                1,ZJC1(25),ZJD1(25),ZJ2(50),ZJC2(25),ZJD2(25),CJ1(50),QJC1(25),QJD100040000
                2(25),QJ2(50),QJC2(25),              FJ1(50),CJ1(50),        CJC1(200050000
                35),CJD1(25),          AJ1(50),AJ2(50),QJC(50),QJD(50),ZJC(50),ZJD(5000060000
                4),VJ(50),EJ1(50),FJ2(50),AJC1(25),AJC2(25),FJC1(27),QJD2(25),ZM2( 00070000
                550),UM2(50),WTC(27)                                                   00080000
0003             COMMON FJ(50),CL(50),CJ(50),AJ(50),XJ(50),YJ(50),                     00090000
                1A(1500),B(1500),C(3000)                                               00100000
0004             COMMON DC,DR,BA,FJ,CC,YN,NN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M           00110000
0005             COMMON CA(50),CB(50),CC(50),CA1(50),QB1(50),QC1(50),QA2(50),QB2(5000120000
                1),CC2(50) ,IH,IE                                                      00130000
0006             COMMON WT(50),HEAD1(15),HEAD2(15)                                     00140000
0007             COMMON/GL/NMC,INT,MMM,IFLT,FN1,FN2,RB1,RD1,RD2,RT1,RT2,RM1,RM2,FI,00150000
                1T1,T2,F1,F2,RF1,RF2,YE1,YE2,CE1,GE2,POS1,POS2,ANG                     00160000
           C SPECIFY AND INITIALIZE   READING AND WRITING UNITS FOR IBM 1800           00170000
0008             COMMON/NEWRAS/EF1,BA1,CC1,XM1,YM1,EP1,DD2,RB2,XM2,YM2,EP2             00180000
0009             COMMON/INH1/ IHICP,UJ(25,4,2),ZJ(25,4,2),WTT(25,4),VPT1,VPT2,CS,     00190000
                1 SLMOJ(25,4),CJJ(25,4,2)                                              00200000
0010             COMMON/CLTH1/ EP12,FF2,W(25,4),VN(25)                                 00210000
0011             COMMON/TAE/ALPH(5,2),Z(5,2),RLOW(2),RHIGH(2)                          00220000
0012             COMMON/INVF/      IFCUR,ISPECT,IRM,IPERC,IHF,IPRCP,ICENT,NZJ          00230000
0013             DATA ZERO/0.0/                                                        00240000
0014             NR=5                                                                  00250000
0015             NW=6                                                                  00260000
0016             FD1=0.                                                                00260100
0017         158 CONTINUE                                                              00270000
0018             RP1=CL*FN1/(FN1+FN2)                                                  00280000
0019             IF(MN.EQ.0)RP1=CL*FN1/(FN2-FN1)                                       00290000
0020             CTT=RB1/RP1                                                           00300000
0021             TTT=SQRT(1.-CTT**2)/CTT                                               00310000
0022             THETP=ATAN(TTT)                                                       00320000
0023             THETPP=THETP*180./3.1415927                                           00330000
0024             WRITE(NW,112)                                                         00340000
0025             WRITE(NW,102) THETPP                                                  00350000
0026             IF(F1-F2) 320,320,321                                                 00360000
0027         321 FF2=F2                                                                00370000
0028             GO TO 322                                                             00380000
0029         320 FF2=F1                                                                00390000
0030         322 FF2=FF2**0.8                                                          00400000
0031         304 CC1=-T1/2.0/RP1+THETP                                                 00410000
0032             F12=FN1/FN2                                                           00420000
0033             RP2=RP1/F12                                                           00430000
0034             CC2=-T2/2.0/RP2+THETP                                                 00440000
0035             FNJ=2.0*F1+1.0                                                        00450000
0036             N=FNJ                                                                 00460000
0037             NN=N+N                                                                00470000
0038             IF(IHICP.NE.0) NN=4*N                                                 00480000
0039             CS=CTT                                                                00490000
0040             TAN=TTT                                                               00500000
0041             EP1=1.0-PCS1*PCS1                                                     00510000
```

```
      EP1=YE1/EP1
      EP2=1.0-FCS2*PCS2
      EP2=YE2/EP2
      EP12=(1.0/EP1+1.0/EP2)*0.5
      EP12=EP12**0.5
      RB2=RB1/F12
      C1=RC1/RB1
      C2=RC2/RB2
      BR2=(SQRT (C2*C2-1.0)-TAN)
      BR1=(SQRT (C1*C1-1.0)-TAN)
      C3=TAN/F12
      IF(BR1-C3) 403,403,404
  404 BR1=C3
      WRITE(NW,126)BR1
  403 EA1=BR2/F12
      EA2=BR1*F12
      IF(KN) 440,450,440
  450 EA1=-BA1
      EA2=-BA2
  440 IF(EA1-TAN) 401,401,402
  402 EA1=TAN
      WRITE(NW,125)EA1
  401 C1=6.2831845/FN1
      IF(EA1-C1) 213,213,611
  611 WRITE(NW,174)
  213 IF(ER1-C1) 215,612,612
  612 WRITE(NW,175)
  215 C1=RT1/RB1
      C2=RT2/RB2
      C1 =SQRT (C1*C1-1.0)-TAN
      AT1=ABS (C1)
      C2 =SQRT (C2*C2-1.0)-TAN
      AT2=ABS (C2)
      WRITE(NW,122)
      WRITE(NW,102)RF1,RB1,EA1,ER1,AT1
      WRITE(NW,162)
      WRITE(NW,102)RF2,RB2,EA2,ER2,AT2
      KMZ=1
      KUZ=1
      IF(AT1-BA1) 211,216,216
  211 WRITE(NW,176)
  216 IF(AT2-BA2) 309,217,217
  309 IF(KN) 620,621,620
  621 WRITE(NW,178)
      GO TO 217
  620 WRITE(NW,179)
  217 IF(RM1-RB1) 191,192,192
  191 AM1=1.0
      WRITE(NW,118)
      GO TO 1920
  192 AM1=RM1/RB1
 1920 C1=AM1
```

```
FORTRAN IV G LEVEL  21                 GEAFC              DATE = 75099          12/49/12

 0094              AM1=SQRT (AM1*AM1-1.0)-TAN                                   01040000
 0095              C2=RF1/RB1                                                   01050000
 0096              FF1=C2                                                       01060000
 0097              C1=RM1/RB1                                                   01070000
 0098              C3=(C1+C2)**2-1.                                             01080000
 0099              IF(C3) 406,406,407                                           01090000
 0100          406 C3=0.0                                                       01100000
 0101              WRITE(NW,156)                                                01110000
 0102              TPD1=0.0                                                     01120000
 0103              GO TO 408                                                    01130000
 0104          407 PD1=SQRT (C3)                                                01140000
 0105              TPD1=ATAN (PD1)                                              01150000
 0106          408 IF(RM2-RF2) 193,194,194                                      01160000
 0107          193 AM2=1.0                                                      01170000
 0108              WRITE(NW,119)                                                01180000
 0109              GO TO 1940                                                   01190000
 0110          194 AM2=RM2/RB2                                                  01200000
 0111         1940 C1=AM2                                                       01210000
 0112              AM2=SQRT (AM2*AM2-1.0)-TAN                                   01220000
 0113              C2=RF2/RB2                                                   01230000
 0114              FF2=C2                                                       01240000
 0115              C1=RM2/RB2                                                   01250000
 0116              IF(MN) 456,455,456                                           01260000
 0117          455 C2=-C2                                                       01270000
 0118          456 C3=(C1+C2)**2-1.0                                            01280000
 0119              IF(C3) 409,409,410                                           01290000
 0120          409 C3=0.0                                                       01300000
 0121              WRITE(NW,155)                                                01310000
 0122              TPD2=0.0                                                     01320000
 0123              GO TO 411                                                    01330000
 0124          410 PD2=SQRT (C3)                                                01340000
 0125              TPD2=ATAN (PD2)                                              01350000
 0126          411 K1=1                                                         01360000
 0127              E1=FF1                                                       01370000
 0128              E2=PD1                                                       01380000
 0129              E3=TPD1                                                      01390000
 0130              E4=RM1                                                       01400000
 0131              D0=D01                                                       01410000
 0132              C1=AM1                                                       01420000
 0133          201 C2=D0+C1                                                     01430000
 0134              C1=ATAN (TAN+C1)                                             01440000
 0135              C4=C1-C2                                                     01450000
 0136              C3=E3-E2-D0+TAN                                              01460000
 0137              IF(K1-1) 422,422,425                                         01470000
 0138          425 IF(MN) 422,421,422                                           01480000
 0139          421 C3=-C3                                                       01490000
 0140              C4=-C4                                                       01500000
 0141          422 C1=C3+E1                                                     01510000
 0142              C4=(C1+C4)*0.5                                               01520000
 0143              Y=E4*SIN (C4)                                                01530000
 0144              X=E4*COS (C4)                                                01540000
 0145              IF(K1-1) 202,202,203                                         01550000
```

```
FORTRAN IV G LEVEL  21                 GEARO              DATE = 75099          12/49/12

        202 XM1=X
            YM1=Y
            K1=K1+1
            CC=CD2
            C1=AM2
            E1=FF2
            E2=FD2
            E3=TPD2
            E4=FM2
            GO TO 201
        203 XM2=X
            YM2=Y
            EE=6.2831854/FNJ
            CC=EE/FN1
            IF(IHICR.NE.0) GO TO 395
            FJ =-FNJ+1.0
            RB=RB1
            ER=ER1
            EA=EA1
            CC=CD1
            XM=XM1
            YM=YM1
            CM=CE1
            F=F1
            II=1
            EP=EP1
            WRITE(NW,151)
            CALL AJCCH
            DO 221 I=1,NN
            C1     =XJ(I)
            C2     =YJ(I)
            FJ1(I)=FJ(I)
            C3=CJ(I)
            AJ1(I)=AJ(I)
            IF(NZJ.EQ.1)
           1CALL TLU(AJ1(I),ZJ1(I),ALPH(1,1),Z(1,1),5)
            CJ1(I)=CJ(I)
            IM1=IM
            IP1=IP
            CA1(I)=CA(I)
            CE1(I)=CE(I)
            CC1(I)=CC(I)
            IF(C3) 515,515,516
        515 WRITE(NW,102)FJ1(I),C3,FJ1(I)
            GO TO 221
        516 WRITE(NW,102)FJ1(I),C3,AJ1(I),CJ1(I),C1,C2
        221 CONTINUE
            WRITE(NW,120)XM1,YM1
            WRITE(NW,161)
            DO 517 I=IM1,IP1
        517 WRITE(NW,102)FJ1(I),QA1(I),QB1(I),QC1(I)
            CC=CD2
```

```
          RB=RB2
          XM=XM2
          YM=YM2
          IF(MN) 302,301,302
      301 FJ=FNJ
          GO TO 310
      302 FJ=-FNJ
      310 F=F2
          EP=EP2
          GM=GE2
          CC=EE/FN2
          II=2
          WRITE(NW,152)
          CALL AJCDH
          L=NN
          DO 222 I=1,NN
          C4     =XJ(I)
          C5     =YJ(I)
          IM2=IM
          IP2=IP
          QA2(I)=QA(I)
          QB2(I)=QB(I)
          QC2(I)=QC(I)
          FJ2(I)=FJ(I)
          C1=AJ(I)
          AJ2(L)=C1
          IF(NZJ.EQ.1)
         1CALL TLU(AJ2(I),ZM2(I),ALFH(1,2),Z(1,2),5)
          C2=CJ(I)
          CJ2(L)=C2
          IF(C2) 519,518,519
      518 WRITE(NW,132)FJ2(I),C1
          GO TO 520
      519 WRITE(NW,132)FJ2(I),C1,C2,C4,C5
      520 IF(L-1) 222,222,226
      226 L=L-1
      222 CONTINUE
          WRITE(NW,120)XM2,YM2
          WRITE(NW,163)
          DO 521 I=IM2,IP2
      521 WRITE(NW,102)FJ2(I),QA2(I),QB2(I),QC2(I)
          DO 251 I=1,N
          IN=I+N
          FJC1(I)=FJ1(IN)
          AJC1(I)=AJ1(IN)
          AJC2(I)=AJ2(IN)
          CJC1(I)=CJ1(I)
          CJC1(I)=CJ1(IN)
          CJC2(I)=CJ2(IN)
          CJC2(I)=CJ2(I)
          CJD1(I)=CJ (I)
      251 CJC1(I)=CJ (IN)
```

FORTRAN IV G LEVEL 21 GEARC DATE = 75099 12/49/12

```
      NUZ=1                                                     02600000
      NWZ=1                                                     02610000
  197 CONTINUE                                                  02620000
      L=NN                                                      02630000
      DO 415 I=1,N                                              02640000
      IN=I+N                                                    02650000
      UJC1(I)=UJ1(I)                                            02660000
      UJC1(I)=UJ1(IN)                                           02670000
      UJC2(I)=UM2(IN)                                           02680000
      UJC2(I)=UM2(I)                                            02690000
      ZJC1(I)=ZJ1(I)                                            02700000
      ZJC1(I)=ZJ1(IN)                                           02710000
      ZJC2(I)=ZM2(IN)                                           02720000
      ZJC2(I)=ZM2(I)                                            02730000
      CJC(I)=CJC1(I)+UJC1(I)+CJC2(I)+UJC2(I)                    02740000
      CJC(I)=CJC1(I)+UJC1(I)+CJC2(I)+UJD2(I)                    02750000
  415 ZJC(I)=ZJC1(I)+ZJC2(I)                                    02760000
  199 CONTINUE                                                  02770000
      WRITE(NW,106)                                             02780000
      WRITE(NW,102)VPT1,VPT2                                    02790000
      VP1=VPT1*CS                                               02800000
      VP2=VPT2*CS                                               02810000
  418 CONTINUE                                                  02820000
      WRITE(NW,115)                                             02830000
      DO 501 I=1,N                                              02840000
      WT(I)=WT(N+1)                                             02850000
      WN=WT(I)/CS                                               02860000
      IF(WN)214,214,800                                         02870000
  800 CONTINUE                                                  02880000
      C1=WN**.1                                                 02890000
      CC=1.37/FF2*EP12/C1                                       02900000
      ZJC(I)=ZJC1(I)+ZJD2(I)+VP1-VP2                            02910000
      CCC    =CJC(I)+CC                                         02920000
      DDD    =CJD(I)+CC                                         02930000
      C1=CJC1(I)                                                02940000
      C2=CJD1(I)                                                02950000
      IF(C1) 256,256,257                                        02960000
  256 IF(C2) 501,501,258                                        02970000
  257 IF(C2) 259,259,260                                        02980000
  259 VJ(I)=ZJC(I)-CCC    *WN                                   02990000
      WTC(I)=WN*CS                                              03000000
      WTC=0.0                                                   03010000
      GO TO 261                                                 03020000
  258 VJ(I)=ZJC(I)-DDD    *WN                                   03030000
      WTC(I)=0.0                                                03040000
      WTC=WN*CS                                                 03050000
      GO TO 261                                                 03060000
  260 C3=CCC    *WN+ZJC(I)-ZJC(I)                               03070000
      C4=DDD    *WN+ZJC(I)-ZJC(I)                               03080000
      IF(C3) 259,259,262                                        03090000
  262 IF(C4) 258,258,263                                        03100000
  263 C1=CCC+DDD                                                03110000
```

```
FCRTRAN IV G LEVEL  21                    GEARC                DATE = 75C99          12/49/12

 C300            C2=CCC    *ZJC(I)+DCD    *ZJC(I)-CCC*DDD*WN
 C301            VJ(I)=C2/C1
 C302            C1=CS/C1
 C303            WTC(I)=(DDC*WN+ZJC(I)-ZJC(I))*C1
 C304            WTD=(CCC*WN+ZJC(I)-ZJC(I))*C1
 C305        261 EJ1(I)=VJ(I)/CS
 C306            AJC1(I)=ANG+AJC1(I)
 C307            WRITE(NW,1C2) FJC1(I),AJC1(I),EJ1(I),ZERO,WTC(I),WTD,ZERC,WN,WT(I)
                1,CC
 C308        501 CCNTINUE
 C309            ANG=AJC1(N)
 C310            GC TO 24
 C311         24 IF(NUZ-NLZ) 504,5C4,502
 C312        502 NLZ=NUZ+1
 C313            GC TO 199
 C314        504 NLZ=1
 C315            IF(NMZ-MMZ) 214,214,506
 C316        506 MMZ=MMZ+1
 C317            GC TO 197
 C318        214 CCNTINUE
 C319        503 RETURN
 C320        399 CALL HICCA
 C321            GO TO 503
 C322        102 FCRMAT(8E13.5)
 C323        106 FCRMAT(//34H INPUT DATA CN TCCTH SPACING ERROR/6X4HVPT19X4HVPT2)
 C324        110 FORMAT(60HCINPUT LISTING CF PRCFILE ERROR AND SUPPLEMENTARY COMPL
                1ANCE/6X2HZ111X2HL111X2HZ211X2HL2)
 C325        112 FORMAT(16HCPRES.ANGLE(DEG))
 C326        115 FCRMAT(42HCCALCULATED TCCTH MESHING ERRORS ANC LOADS    //
                1 57X,15HTANGENTIAL LCAC/
                2 6X,3HJC1,5X,4HAJC1,6X,11HTANG. ERRCR,4X,7HE2 PAIR,6X,7HE1 PAIR,
                3  6X,7H C PAIR,6X,7H-1 PAIR,8X,2HWN,11X,2HWT,10X,3HQJD)
 C327        118 FCRMAT(74HCDRIVING GEAR INPUT ROOT RADIUS SMALLER THAN BASE CIRCLE
                1 RADIUS                          )
 C328        119 FCRMAT(74HCDRIVEN  GEAR INPUT ROOT RADIUS SMALLER THAN BASE CIRCLE
                1 RADIUS                          )
 C329        120 FCRMAT(48HCCCCC. CF EFFECTIVE TOOTH PROFILE AT ROOT CIRCLE11X1HX13
                1X1HY/52X,2E13.6)
 C330        122 FCRMAT(/6X3HRP11CX3HRB110X3HEA110X3HBR110X3HAT1)
 C331        125 FCRMAT(66HC DRIVEN GEAR TEETH ENGAGE UNDER CUT PORTION CF DRIVING
                1GEAR TEETH//6H  EA1=,  E13.6)
 C332        126 FCRMAT(66HC DRIVING GEAR TEETH ENGAGE UNDER CUT PORTION OF DRIVEN
                1GEAR TEETH//6H  EE1=,  E13.6)
 C333        132 FCRMAT(  E13.5,13X,4E13.5)
 C334        151 FCRMAT(/6X2HJ111X3HCJ110X3HAJ18X6HQJ1ABC9X3HXJ110X3HYJ1)
 C335        152 FCRMAT(/6X2HJ224X3HAJ28X6HQJ2ABC9X3HXJ210X3HYJ2)
 C336        155 FCRMAT(63HCDRIVEN  GEAR INPUT RADIUS TO FILLET CENTER INSIDE BASE
                1CIRCLE./42HCPRCGRAM CCNTINUES WITH CORRECT TREATMENT.)
 C337        156 FCRMAT(63HCDRIVING GEAR INPUT RADIUS TO FILLET CENTER INSIDE BASE
                1CIRCLE./42HCPRCGRAM CCNTINUES WITH CORRECT TREATMENT.)
 C338        159 FCRMAT(13X,4E13.5)
 C339        161 FCRMAT(/6X2HJ11CX4HCJ1ASX4HQJ1B9X4HQJ1C)
```

```
C340        162 FORMAT(/6X3HRP210X3HRB210X3HPA210X3HBA210X3HAT2)                 C3640000
C341        163 FORMAT(/6X2HJ210X4HQJ2A5X4HQJ2B9X4HQJ2C)                         C3650000
C342        170 FORMAT(43HCCALCULATED TOTAL CONTACT COMPLIANCE   QJD#,E13.6)     C3660000
C343        171 FORMAT(23HCCALCULATED NORMAL  WN#.  E13.6)                       03670000
C344        174 FORMAT(8CHC ANGLE CF APPRCACH CN DRIVING GEAR IS GREATER THAN TOOTC3680000
               1H SPACING ANGLE.                   /                             C3690000
               2        40H  PROGRAM CONTINUED WITHOUT OVERLAP                   )03700000
C345        175 FORMAT(75HC ANGLE CF RECESS CN DRIVING GEAR IS NOT SMALLER THAN TCC3710000
               10TH SPACING ANGLE. /34H PROGRAM CONTINUED WITHOUT OVERLAP)       C3720000
C346        176 FORMAT(8CHC DRIVING GEAR TEETH MESHING ON PROFILE INSIDE OF TIF D10373C000
               1AMETER,                                    /                     C3740000
               2        40H  PROGRAM CONTINUED WITHOUT CORRECTION                )C3750000
C347        178 FORMAT(8CHC DRIVEN  GEAR TEETH MESHING ON PROFILE OUTSIDE OF TIF D03760000
               1IAMETER.                                   /                     C3770000
               2        40H  PROGRAM CONTINUED WITHOUT CORRECTION                )C378C000
C348        179 FORMAT(8CHC DRIVEN  GEAR TEETH MESHING CN PROFILE INSIDE OF TIF DIC3790000
               1AMETER. PROGRAM CONTINU-/21HED WITHOUT CORRECTION)               C3800000
C349            END                                                              C3810000
```

```
FORTRAN IV G LEVEL  21                 HICON               DATE = 75099          12/49/12

 0001               SUBROUTINE HICON                                                  00010000
            C**** MEMBER NAME R75HICON                                                00020000
 0002               DIMENSION ANG1(25,4)                                              00020100
 0003               COMMON PJ(50),CJ(50),QJ(50),AJ(50),XJ(50),YJ(50),                 00030000
                   1 A(1500),E(1500),C(3000)                                          00040000
 0004               COMMON DD,ER,BA,FJ,CC,YM,NN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M       00050000
 0005               COMMON CA(50),CB(50),CC(50),QA1(50),QB1(50),QC1(50),QA2(50),QB2(5000060000
                   1),QC2(50),JM,IP                                                   00070000
 0006               COMMON WT(50),HEAD1(15),HEAD2(15)                                 00080000
 0007               COMMON/HICC/GNA(100),GNB(100),GNC(100),GNJ(100),XNJ(100),         00090000
                   1  YNJ(100), ANGLE(100),PNJ(100),CONJ(100),CONCC(100),SND(100),    00100000
                   2  CND(100),EINK(100),AINK(100),FLAK(100)                          00110000
 0008               COMMON/NEWEAS/ER1,BA1,CD1,XM1,YM1,EP1,DD2,RB2,XM2,YM2,EP2         00120000
 0009               COMMON/CL/NMC,INT,MMM,IFL1,FN1,FN2,RB1,RC1,RD2,RT1,RT2,RM1,RM2,F1,00130000
                   1T1,T2,F1,F2,RF1,RF2,YE1,YE2,GE1,GE2,POS1,POS2,ANG                 00140000
 0010               COMMON/INF1/ IFICR,UJ(25,4,2),ZJ(25,4,2),WTT(25,4),VPT1,VPT2,CS,  00150000
                   1 SUMCJ(25,4),CJJ(25,4,2)                                          00160000
 0011               COMMON/CLTH/ EF12,FF2,W(25,4),VN(25)                              00170000
 0012               COMMON/YY/ CUM1(1000),E 1(56),FJ2(50),AJC1(25),CUM2(204)          00180000
 0013               COMMON/TAB/ ALPH(5,2),Z(5,2),FLOW(2),RHIGH(2)                     00190000
 0014               COMMON/IAVR/      IFOUR,ISPECT,IRM,IPERC,IRF,IPROP,ICENT,NZJ      00200000
 0015               COMMON/DIAG/ IDIAG                                                00210000
 0016               IF(IDIAG.GT.0)                                                    00230000
                   1WRITE(6,1010) ZJ                                                  00240000
 0017               IF(IDIAG.GT.0)                                                    00250000
                   1WRITE(6,1000) CJJ                                                 00260000
 0018               NW=6                                                              00270000
 0019               FJ=-2.*FNJ+1                                                      00280000
 0020               RB=RB1                                                            00290000
 0021               ER=ER1                                                            00300000
 0022               BA=BA1                                                            00310000
 0023               DD=DD1                                                            00320000
 0024               XM=XM1                                                            00330000
 0025               YM=YM1                                                            00340000
 0026               GM=GE1                                                            00350000
 0027               F=F1                                                              00360000
 0028               II=1                                                              00370000
 0029               EP=EP1                                                            00380000
 0030               WRITE(NW,151)                                                     00390000
 0031               CALL NAME                                                         00400000
 0032               K=0                                                               00410000
 0033               DO 221 I=1,4                                                      00420000
 0034               DO 221 J=1,N                                                      00430000
 0035               K=K+1                                                             00440000
 0036               ANG1(J,I)=ANGLE(K)                                                00450000
 0037               IF(NZJ.EQ.1)                                                      00460000
                   1CALL TLU(ANGLE(K),ZJ(J,I,2),ALPH(1,1),Z(1,1),5)                   00470000
 0038               IF(CONJ(K)  .EQ.0.) WRITE(NW,102) PNJ(K)  ,CONJ(K)   ,ANGLE(K)    00480000
 0039               IF(CONJ(K)  .EQ.1.) WRITE(NW,102) PNJ(K)  ,CONJ(K)   ,ANGLE(K)    00490000
                   1 GNJ(K)  ,XNJ(K)  ,YNJ(K)  ,ZJ(J,I,1)                             00500000
 0040               CJJ(J,I,1)=GNJ(K)                                                 00510000
 0041           221 CONTINUE                                                          00520000
```

```
FORTRAN IV G LEVEL  21                MICCN              DATE = 75099           12/49/12

C042            WRITE(NW,120) XM1,YM1                                             00530000
C043            WRITE(NW,101)                                                     00540000
C044            DO 517 K=1,NN                                                     00550000
C045            IF(CCNJ(K)  .EQ.0.) GO TO 517                                     00560000
C046            WRITE(NW,102) FNJ(K),CNA(K),CNB(K),CNC(K)                         00570000
C047        517 CONTINUE                                                          00580000
C048            CC=CC2                                                            00590000
C049            RB=RB2                                                            00600000
C050            XM=XM2                                                            00610000
C051            YM=YM2                                                            00620000
C052            FJ=FNJ*2.                                                         00630000
C053            IF(MN.NE.0) FJ=-FNJ*2.                                            00640000
C054            F=F2                                                              00650000
C055            EP=EP2                                                            00660000
C056            GM=GE2                                                            00670000
C057            CC=CC*FN1/FN2                                                     00680000
C058            II=2                                                              00690000
C059            WRITE(NW,152)                                                     00700000
C060            CALL NANG                                                         00710000
C061            K=0                                                               00720000
C062            LI=4+1                                                            00730000
C063            DO 222 I=1,4                                                      00740000
C064            LJ=N+1                                                            00750000
C065            LI=LI-1                                                           00760000
C066            DO 222 J=1,N                                                      00770000
C067            K=K+1                                                             00780000
C068            LJ=LJ-1                                                           00790000
C069            CJJ(LJ,LI,2)=CNJ(K)                                               00800000
C070            IF(NZJ.EQ.1)                                                      00810000
               1CALL TLL(ANGLE(K),ZJ(LJ,LI,2),ALPH(1,2),Z(1,2),5)                 00820000
C071            IF(CONDC(K)  .EQ.0.) WRITE(NW,132) PNJ(K)   ,ANGLE(K)             00830000
C072            IF(CONDC(K)  .EQ.1.) WRITE(NW,132) PNJ(K)   ,ANGLE(K)   ,CNJ(K)  ,00840000
               1   XNJ(K)  ,YNJ(K) ,ZJ(LJ,LI,2)                                   00850000
C073        222 CONTINUE                                                          00860000
C074            WRITE(NW,120) XM2,YM2                                             00870000
C075            WRITE(NW,103)                                                     00880000
C076            DO 521 K=1,NN                                                     00890000
C077            IF(CONDC(K)  .EQ.0.) GO TO 521                                    00900000
C078            WRITE(NW,102) FNJ(K),CNA(K),CNB(K),CNC(K)                         00910000
C079        521 CONTINUE                                                          00920000
C080            IF(IDIAG.GT.0)                                                    00930000
               1WRITE(6,1010) ZJ                                                  00940000
C081            IF(IDIAG.GT.0)                                                    00950000
               1WRITE(6,1000) CJJ                                                 00960000
C092       1000 FORMAT(2H C,(10E13.5))                                            00970000
C093       1010 FORMAT(2H Z,10E13.5)                                              00980000
C094            CALL SOLMA                                                        00990000
C095            WRITE(NW,115)                                                     01000000
C096            DO 501 I=1,N                                                      01010000
C087            XI=I                                                              01020000
C088            EJ1(I)=VN(I)/CS                                                   01030000
C089            WN=WTT(I,3)/CS                                                    01040000
```

```
FORTRAN IV G LEVEL  21                  HICCA              DATE = 75099          12/49/12

 0090          CD=1.37/FF2*EF12/WN**.1                                             01050000
 0091          AJC1(I)=ANG1(I,2)+ANG                                               01060000
 0092      501 WRITE(NW,102)     XI,A.C1(I),EJ1(I),(W(I,J),J=1,4),WN,WTT(I,3),QD01070000
 0093          ANG=AJC1(N)                                                         01080000
 0094          RETURN                                                              01090000
 0095      102 FORMAT(10E13.5)                                                     01100000
 0096      115 FORMAT(42HCCALCULATED TOOTH MESHING ERRORS AND LOADS  //            01110000
              1 57X,15HTANGENTIAL LOAD/                                            01120000
              2  6X,3HJC1,5X,4HAJC1,6X,11HTANG. ERROR,4X,7HE2 PAIR,6X,7HE1 PAIR,  01130000
              3  6X,7H C PAIR,6X,7H-1 PAIR,8X,2HWN,11X,2HWT,10X,3HQJD)             01140000
 0097      120 FORMAT(48HCCOCCC. OF EFFECTIVE TOOTH PROFILE AT ROOT CIRCLE         01150000
              111X,1HX,13X,1HY,/,52X,2E13.6)                                       01160000
 0098      132 FORMAT(E13.5,12X,5E13.5)                                            01170000
 0099      151 FORMAT(/6X,2HJ1,11X,3HCJ1,10X,3HAJ1,8X,6HCJ1ABC,9X,3HXJ1,10X,3HYJ101180000
              1,10X,3HZJ1)                                                         01190000
 0100      152 FORMAT(/6X,2HJ2,24X,3HAJ2,8X,6HCJ2ABC,9X,3HXJ2,10X3HYJ2,10X,3HZJ2)01200000
 0101      161 FORMAT(/ 6X,2HJ1,10X,4HCJ1A,9X,4HCJ1B,9X,4HQJ1C)                    01210000
 0102      163 FORMAT(/6X, 2HJ2,10X,4HCJ2A,9X,4HCJ2B,9X,4HQJ2C)                    01220000
 0103          END                                                                 01230000
```

```
FORTRAN IV G LEVEL  21                  MATINV              DATE = 75C99          12/49/12

C001             SUBROUTINE MATINV (A,N,B,M,DETER)
           C**** MEMBER NAME R75MATIN
           C     THIS IS THE SHORT VERSION OF MATINV, IT SOLVES A SYSTEM OF
           C     EQUATIONS WITHOUT GETTING THE INVERSE OF THE A MATRIX
           C     THEREFORE THIS VERSION SHOULD NOT BE USED WITH THE INTENT TO
           C     OBTAIN THE INVERSE OF A, USE THE LONG VERSION FOR THIS PURPOSE
           C     NOTE ALSO THAT THE CONSTANT OR SOLUTION VECTOR IS A ONE
           C     DIMENSIONAL ARRAY
C002             DIMENSION A(5,5),B(5),IPIVO(5),PIVOT(5)
C003             DETER =1.0
C004             DO 20 J=1,N
C005          20 IPIVO(J)=0
C006             DO 550 I=1,N
           C     SEARCH FOR PIVOT ELEMENT
C007             AMAX=0.0
C008             DO 105 J=1,N
C009             IF (IPIVO(J)-1) 60,105,60
C010          60 DO 100 K=1,N
C011             IF (IPIVO(K) -1) 80, 100, 60C
C012          80 IF (ABS (AMAX)-ABS (A(J,K))) 85,100,100
C013          85 IROW=J
C014             ICOLU =K
C015             AMAX=A(J,K)
C016         100 CONTINUE
C017         105 CONTINUE
C018             IPIVO(ICOLU)=IPIVO(ICOLU)+1
           C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
C019             IF (IROW-ICOLU)  140, 260, 140
C020         140 DETER =-DETER
C021             DO 200 L=1,N
C022             AMAX=A(IROW,L)
C023             A(IROW,L)=A(ICOLU,L)
C024         200 A(ICOLU,L)=AMAX
C025             AMAX=B(IROW)
C026             B(IROW)=B(ICOLU)
C027             B(ICOLU)=AMAX
C028         260 PIVOT(I)=A(ICOLU,ICOLU)
C029             DETER =DETER*PIVOT(I)
           C     DIVIDE PIVOT ROW BY PIVOT ELEMENT
C030             A(ICOLU,ICOLU)=1.0
C031             DO 350 L=1,N
C032         350 A(ICOLU,L)=A(ICOLU,L)/PIVOT(I)
C033             B(ICOLU)=B(ICOLU)/PIVOT(I)
           C     REDUCE NON-PIVOT ROWS
C034         380 DO 550 L1=1,N
C035             IF(L1-ICOLU)  400, 550, 400
C036         400 AMAX=A(L1,ICOLU)
C037             A(L1,ICOLU) =0.0
C038             IF(AMAX)430,550,430
C039         430 DO 450 L=1,N
C040         450 A(L1,L)=A(L1,L)-A(ICOLU,L)*AMAX
C041             B(L1)=B(L1)-B(ICOLU)*AMAX
```

```
FORTRAN IV G LEVEL  21                  MATINV              DATE = 75099          12/49/12

 0042          550 CONTINUE                                                       00530000
 0043          600 RETURN                                                         00540000
 0044              END                                                            00550000
```

```
FORTRAN IV G LEVEL  21                    NANG             DATE = 75099          12/49/12

 0001               SUBROUTINE NANG                                              00010000
              C**** MEMBER NAME R75NANG                                          00020000
 0002               COMMON PJ(50),CJ(50),GJ(50),AJ(50),XJ(50),YJ(50),            00030000
                   1 A(1500),E(1500),G(3000)                                     00040000
 0003               COMMON DD,ER,BA,FJ,CC,YM,NN,N,FNJ,EF,TAN,RB,GM,F,XM,MN,II,M  00050000
 0004               COMMON CA(50),CB(50),CC(50),CA1(50),QB1(50),QC1(50),QA2(50),QB2(5000060000
                   1),CC2(50),IH,IF                                              00070000
 0005               COMMON/FICC/CNA(100),CNB(100),CNC(100),GNJ(100),XNJ(100),    00080000
                   1 YNJ(100),ANJ(100),PNJ(100),CNJ(100),CNDC(100),SND(100),CND(100),00090000
                   2 BINK(100),AINK(100),FLAK(100)                               00100000
 0006               DO 106 I=1,NN                                                00110000
 0007               CNA(I)=0.0                                                   00120000
 0008               CNB(I)=0.0                                                   00130000
 0009               CNC(I)=0.0                                                   00140000
 0010               CNJ(I)=0.0                                                   00150000
 0011               XNJ(I)=0.0                                                   00160000
 0012           106 YNJ(I)=0.0                                                   00170000
 0013               IF(II-1) 405,405,403                                         00180000
 0014           405 MM=1.0                                                       00190000
 0015               GO TO 404                                                    00200000
 0016           403 MM=MN                                                        00210000
 0017           404 K=1                                                          00220000
 0018               DO 501 I=1,NN                                                00230000
 0019               ANJ(I)=FJ*CC                                                 00240000
 0020               FNJ(I)=FJ                                                    00250000
 0021               IF(II-1) 412,412,413                                         00260000
 0022           412 CALL CALCJ(ANJ(I),ER,BA,XY)                                  00270000
 0023               CNJ(I)=XY                                                    00280000
 0024               GO TO 414                                                    00290000
 0025           413 XY=CNDC(I)                                                   00300000
 0026           414 IF(XY) 503,503,101                                           00310000
 0027           101 IF(K-1) 102,102,103                                          00320000
 0028           102 IH=I                                                         00330000
 0029           103 AX=ANJ(I)                                                    00340000
 0030               DJ=DD+AX                                                     00350000
 0031               SND(I)=SIN(DJ)                                               00360000
 0032               CND(I)=COS(DJ)                                               00370000
 0033               C1=ATAN (TAN+AX)                                             00380000
 0034               C3=COS (C1)                                                  00390000
 0035               C4=C1-DJ                                                     00400000
 0036               IF(MM) 402,401,402                                           00410000
 0037           401 C4=-C4                                                       00420000
 0038           402 C5=SIN (C4)                                                  00430000
 0039               C6=COS (C4)                                                  00440000
 0040               C1=RB/C3                                                     00450000
 0041               XNJ(I)=C1*C6                                                 00460000
 0042               YNJ(I)=C1*C5                                                 00470000
 0043               IF(K-1) 242,242,243                                          00480000
 0044           242 BINK(I)=(YNJ(I)**3+YM**3)/3.0*F                              00490000
 0045               AINK(I)=(YNJ(I)+YM)*F                                        00500000
 0046               FLAK(I)=XNJ(I)-XM                                            00510000
 0047               GO TO 504                                                    00520000
```

```
  243 EINK(I)=(YNJ(I)**3+YNJ(I-1)**3)/3.0*F
      AINK(I)=(YNJ(I)+YNJ(I-1))*F
      FLNK(I)=XNJ(I)-XNJ(I-1)
  504 K=K+1
  503 IF(I-NN) 502,501,501
  502 IF(MM) 506,505,506
  505 FJ=FJ-1.
      GO TO 501
  506 FJ=FJ+1.0
  501 CONTINUE
      IF(II-1) 301,301,302
  301 IF(MM) 601,302,601
  601 LL=NN
      DO 303 I=1,NN
      CNDC(I)=CNJ(LL)
      IF(LL-1) 303,303,305
  305 LL=LL-1
  303 CONTINUE
  302 IP=K+IM-2
  410 DO 246 L=IF,IP
      E1=0.0
      E2=0.0
      E3=0.0
      E4=0.0
      EE=XNJ(L)-XM
      IM=L
      IF(MM) 202,203,202
  203 EE=-EE
  202 DO 245 I=IF,IM
      C1=XNJ(L)-XNJ(I)
      C5=FLNK(I)
      IF(MM) 207,208,207
  208 C1=-C1
      C5=-C5
  207 C2=C5/BINK(I)
      C3=(C5+3.0*C1)*C5+3.0*C1*C1
      E4=E4+C2
      C4=2.0*C1+C5
      E1=E1+C3*C2
      E2=E2+C4*C2
  245 E3=E3+C5/AINK(I)
      IF(MM) 206,205,206
  205 IM=IM-1
  206 C1=CND(L)*CNC(L)
      C2=SND(L)*YNJ(L)
      CNA(L) =C1/3.0*E1/EP+(C2*E4-CND(L)*E2)/EP*C2
      CNB(L) =C1/GM*E3+1.2
      C1=(EE*CND(L)-YNJ(L)*SND(L))/YM
      CNC(L)=C1/EP*1.327/F*C1
  246 CNJ(L)=CNA(L)+CNB(L)+CNC(L)
      RETURN
      END
```

```
FORTRAN IV G LEVEL  21                 PLT                 DATE = 75099          12/49/12

 0001              SUBROUTINE PLT(X,Y,Z,NPTS)                                       00010000
            C****  MEMBER NAME R75PLT                                               00020000
 0002              DIMENSION IBUF(4000)                                             00020100
 0003              REAL*4 ILABX(6),ILABY(6),ILABZ(3)                                00020200
 0004              DIMENSION Y(3000),X(3000),Z(3000),AXX(4),AYY(4),AZZ(4),HEAD3(4)  00030000
 0005              COMMON PJ(50),CJ(50),QJ(50),AJ(50),XJ(50),YJ(50),                00050000
                  1A(1500),B(1500),C(3000)                                          00060000
 0006              COMMON CC,ER,RA,FJ,CO,YM,AN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M      00070000
 0007              COMMON CA(50),CE(50),CC(50),CA1(50),QB1(50),QC1(50),QA2(50),QB2(5000080000
                  1),CC2(50) ,IH,IF                                                 00090000
 0008              COMMON WT(50),HEAD1(15),HEAD2(15)                                00100000
 0009              DATA ILABX/'    ','CALC','ULAT','ION ','PGIN','T   '/             00110000
 0010              DATA ILABY/'TANG','ENTI','AL E','RROR',' (IN','.)  '/             00120000
 0011              DATA ILABZ/'LOAD',' (LB','.)  '/                                 00130000
 0012              DATA HEAD3/'TANC','. FO','RCE ','  = '/                          00140000
 0013              DATA EHEAD3/'LB. '/                                              00140100
 0014              IN=5                                                             00150000
 0015              IO=6                                                             00160000
 0016              ICAL=22                                                          00170000
 0017              CALL PLOTS(IBUF,4000,ICAL)                                       00180000
 0018              CALL PLOT(4.,4.5,-3)                                             00190000
 0019              CALL PLOT(-2.,-4.,3)                                             00200000
 0020              CALL DASHP(-2.,6.,.25)                                           00210000
 0021              CALL SCALE(X,5.,NPTS,+1)                                         00220000
 0022              Y(NPTS+1)=0.                                                     00230000
 0023              DO 10 K=1,NPTS                                                   00240000
 0024           10 Y(K)=-Y(K)                                                       00250000
 0025              CALL SCALE(Y,4.,NPTS+1,+1)                                       00260000
 0026              Y(NPTS+2)=Y(NPTS+3)                                              00270000
 0027              CALL SCALE(Z,3.,NPTS,+1)                                         00280000
 0028              AXX(1)=X(NPTS+1)                                                 00290000
 0029              AXX(2)=X(NPTS+2)                                                 00300000
 0030              AYY(1)=Y(NPTS+1)                                                 00310000
 0031              AYY(2)=Y(NPTS+2)                                                 00320000
 0032              AZZ(1)=Z(NPTS+1)                                                 00330000
 0033              AZZ(2)=Z(NPTS+2)                                                 00340000
 0034              CALL AXIS(0.,0.,ILABX,-24,5.,0.,AXX(1),AXX(2))                   00350000
 0035              CALL AXIS(0.,0.,ILABY,24,4.,90.,0.,AYY(2))                       00360000
 0036              CALL FLINE(X,Y,-NPTS,1,1,3)                                      00370000
 0037              CALL PLOT(0.,-4.,-3)                                             00380000
 0038              CALL AXIS(0.,0.,ILABX,-24,5.,0.,AXX(1),AXX(2))                   00390000
 0039              CALL AXIS(0.,0.,ILABZ,12,3.,90.,AZZ(1),AZZ(2))                   00400000
 0040              CALL FLINE(X,Z,-NPTS,1,1,11)                                     00410000
 0041              CALL PLOT(.5,9.5,-3)                                             00420000
 0042              CALL SYMBOL(0.,0.,.14,HEAD1,0.,40)                               00430000
 0043              CALL SYMBOL(0.,-.25,.14,HEAD2,0.,40)                             00440000
 0044              CALL SYMBOL(0.,-.5,.14,HEAD3,0.,16)                              00450000
 0045              CALL NUMBER(999.,999.,.14,WT,0.,2)                               00460000
 0046              CALL SYMBOL(999.,999.,.14,EHEAD3,0.,4)                           00470000
 0047              CALL PLOT(6.,-9.,-3)                                             00480000
 0048              CALL DASHP(0.,9.,.25)                                            00490000
 0049              CALL PLOT(0.,0.,999)                                             00500000
```

```
FORTRAN IV G LEVEL  21                PLT               DATE = 75099          12/49/12

0050               DO 20 K=1,NPTS                                              00510000
0051            20 Y(K)=-Y(K)                                                  00520000
0052               RETURN                                                      00530000
0053               END                                                         00540000
```

```
FORTRAN IV G LEVEL  21                    SOLMA              DATE = 75099          12/49/12

 C001                SUBROUTINE SOLMA                                                C0010000
               C**** MEMBER NAME R75SOLMA                                            C0020000
 C002                COMMON PJ(50),CJ(50),QJ(50),AJ(50),XJ(50),YJ(50),               C0030000
                    1 A(1500),E(1500),G(3000)                                        C0040000
 C003                COMMON DC,ER,BA,FJ,CC,YM,NN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M     C0050000
 C004                COMMON QA(50),CB(50),CC(50),QA1(50),QB1(50),QC1(50),QA2(50),QB2(50C0060000
                    1),QC2(50),IH,IF                                                 C0070000
 C005                COMMON WT(50),HEAD1(15),HEAD2(15)                               C0080000
 C006                COMMON/OUTPT/ EF12,FF2,W(25,4),VN(25)                           C0090000
 C007                COMMON/INHI/ IHICR,UJ(25,4,2),ZJ(25,4,2),WTT(25,4),VPT1,VPT2,CS, C0100000
                    1 SUMQJ(25,4),QJJ(25,4,2)                                        C0110000
 C008                COMMON/DIAG/ IDIAG                                              C0120000
 C009                DIMENSION AA(5,5),INDEX(5),RHS(5)                               C0130000
 C010                VP1=VPT1*CS                                                     C0140000
 C011                VP2=VPT2*CS                                                     C0150000
 C012                DO 100 J=1,N                                                    C0160000
 C013              5 CONTINUE                                                        C0170000
               C   *  ZERO OUT AA MATRIX                                             C0180000
 C014                DO 10 IM=1,5                                                    C0190000
 C015                DO 10 JM=1,5                                                    C0200000
 C016                AA(IM,JM)=0.0                                                   C0210000
 C017             10 CONTINUE                                                        C0220000
               C   *  SET COUNTER IC=0                                               C0230000
 C018                IC=0                                                            C0240000
               C   *  START TOOTH LOOP                                               C0250000
 C019                WN=WTT(J,3)/CS                                                  C0260000
 C020                QJ120=1.37/FF2*EF12/WN**1.1                                     C0270000
 C021                DO 25 K=1,4                                                     C0280000
 C022                RSIDE=ZJ(J,K,1)+ZJ(J,K,2) +(VP1-VP2)*FLOAT(3-K)                 C0290000
 C023                IF(IDIAG.GT.0)                                                  C0300000
                    1WRITE(6,1000)J,K,ZJ(J,K,1),ZJ(J,K,2),QJJ (J,K,1),QJJ(J,K,2),RSIDE,C0310000
                    1 QJ12,SUMQJ(J,K),WTT(J,3)                                       C0320000
 C024           1000 FORMAT(2H 5,2I3,8E13.5)                                         C0330000
 C025                IF(QJJ(J,K,1).EQ.0.0.AND.QJJ(J,K,2).EQ.0.0) GO TO 25            C0340000
 C026                QJ12=ABS(QJJ(J,K,1))+ABS(QJJ(J,K,2))+ABS(QJ120)                 C0350000
 C027                SUMQJ(J,K)=QJ12+UJ(J,K,1)+UJ(J,K,2)                             C0360000
 C028                IC=IC+1                                                         C0370000
 C029                INDEX(IC)=K                                                     C0380000
 C030                AA(IC,IC)=SUMQJ(J,K)                                            C0390000
 C031                RHS(IC)=RSIDE                                                   C0400000
 C032             25 CONTINUE                                                        C0410000
 C033                DO 30 L=1,IC                                                    C0420000
 C034                AA(L,IC+1)=1.                                                   C0430000
 C035                AA(IC+1,L)=1.                                                   C0440000
 C036                RHS(IC+1)=WN                                                    C0450000
 C037             30 CONTINUE                                                        C0460000
 C038                CALL MATINV(AA,IC+1,RHS,1,DETER,ID)                             C0470000
 C039                IZERO=0                                                         C0480000
 C040                DO 50 K=1,IC                                                    C0490000
 C041                ISLB=INDEX(K)                                                   C0500000
 C042                IF(RHS(K).GT.0.0) GO TO 50                                      C0510000
 C043                QJJ(J,ISLB,1)=0.0                                               C0520000
```

```
FORTRAN IV G LEVEL  21                    SOLVA              DATE = 75099            12/49/12
                                                                                     00530000
 0044              CJJ(J,ISUB,2)=0.0                                                00540000
 0045              IZERO=IZERO+1                                                     00550000
 0046         50   CONTINUE                                                          00560000
 0047              IF(IZERO.NE.0) GO TO 5                                            00570000
 0048              DO 70 K=1,IC                                                      00580000
 0049              ISUB=INDEX(K)                                                     00590000
 0050              W(J,ISUB)=RHS(K)*CS                                               00600000
 0051         70   CONTINUE                                                          00610000
 0052              VN(J)=RHS(IC+1)                                                   00620000
 0053        100   CONTINUE                                                          00630000
 0054              RETURN                                                            00640000
 0055              END
```

```
FORTRAN IV G LEVEL  21                SPECT              DATE = 75099          12/49/12

 0001               SUBROUTINE SPECT                                                   00010000
              C**** MEMBER NAME R75SPECT                                               00020000
 0002               DIMENSION CN(3000),RR(3000),FR(200),GK(200),KK(200),GKK(200)       00030000
 0003               COMMON PJ(50),CJ(50),QJ(50),AJ(50),XJ(50),YJ(50),                  00040000
                   1A(1500),B(1500),G(2000)                                            00050000
 0004               COMMON DC,PR,PA,FJ,CC,YM,MM,N,FNJ,EF,TAN,RB,GM,F,XM,MN,II,M        00060000
 0005               COMMON/CL/NMC,INT,MMM,IFL1,FN1,FN2,RB1,RO1,RO2,RT1,RT2,RM1,RM2,FI,00070000
                   1T1,T2,F1,F2,RF1,RF2,YE1,YE2,GE1,GE2,POS1,POS2,ANG                  00080000
 0006               NR=5                                                               00090000
 0007               NW=6                                                               00100000
              C NWS=1,WS=SPEED OF DRIVING GEAR IN RPM                                  00110000
              C NWS=2,WS=SPEED OF DRIVEN GEAR IN RPM                                   00120000
 0008               READ(NR,100) NWS,WS                                                00130000
 0009               WS=WS*6.28/60.                                                     00140000
 0010               FM=FN1*WS/6.28                                                     00150000
 0011               IF(NWS.EQ.2) FM=FN2*WS/6.28                                        00160000
 0012               WRITE(NW,110) FM                                                   00170000
 0013               FC=FM*N/2.                                                         00180000
 0014               NT=N*NMC                                                           00190000
 0015               TR=NMC/FM                                                          00200000
 0016               H=1./FM/N                                                          00210000
 0017               FC=1./TR                                                           00220000
 0018               M=NT/20                                                            00230000
 0019               SM=M                                                               00240000
 0020               TMAX=M*H                                                           00250000
 0021               BE=1./TMAX                                                         00260000
 0022               WRITE(NW,115)                                                      00270000
 0023               WRITE(NW,120) FC,FC,BE                                             00280000
 0024               WRITE(NW,125)                                                      00290000
 0025               WRITE(NW,120) H,TR,TMAX,SM                                         00300000
 0026               SUM=0.                                                             00310000
 0027               DO 10 I=1,NT                                                       00320000
 0028            10 SUM=SUM+G(I)                                                       00330000
 0029               SA=SUM/NT                                                          00340000
 0030               DO 20 I=1,NT                                                       00350000
 0031               CN(I)=G(I)-SA                                                      00360000
 0032            20 CONTINUE                                                           00370000
 0033               MM=M+1                                                             00380000
 0034               DO 30 I=1,MM                                                       00390000
 0035               IR=I-1                                                             00400000
 0036               FR(I)=IR*FC/SM                                                     00410000
 0037               CC=1./(NT-IR)                                                      00420000
 0038               RR(I)=0.                                                           00430000
 0039               DO 25 J=1,NT                                                       00440000
              C RR$I<$R$I-1<                                                           00450000
 0040               IF((J+IR).GT.NT) GO TO 30                                          00460000
 0041            25 RR(I)=CN(J)*CN(J+IR)+RR(I)                                         00470000
 0042            30 RR(I)=RR(I)*CC                                                     00480000
 0043               MS=M-1                                                             00490000
 0044               DO 40 K=1,MM                                                       00500000
 0045               KK(K)=K-1                                                          00510000
 0046               GK(K)=RR(1)+RR(MM)*(-1.)**KK(K)                                    00520000
```

```
FORTRAN IV G LEVEL  21                    SPECT              DATE = 75099          12/49/12

                    SSS=0.                                                              00530000
                    DO 35 JK=1,MS                                                       00540000
                    JJ=JK+1                                                             00550000
                    AA=3.1416*JK*KK(K)/SM                                               00560000
               35   SSS=SSS+2.*R(JJ)*COS(AA)                                            00570000
               40   GK(K)=(CK(K)+SSS)*2.*H                                              00580000
                    GKK(1)=.5*GK(1)+.5*GK(2)                                            00590000
                    GKK(MM)=.5*GK(M)+.5*GK(MP)                                          00600000
                    DO 50 LL=2,M                                                        00610000
                    GKK(LL)=.25*GK(LL-1)+.5*GK(LL)+.25*GK(LL+1)                         00620000
               50   CONTINUE                                                            00630000
                    WRITE(NW,130)                                                       00640000
                    WRITE(NW,140)                                                       00650000
                    DO 60 I=1,MM                                                        00660000
                    SKM=KK(I)*N/(2.*SM)                                                 00670000
                    WRITE(NW,150)KK(I),FR(I),GK(I),GKK(I),SKM                           00680000
               60   CONTINUE                                                            00690000
              100   FORMAT(I5,E13.5)                                                    00700000
              110   FORMAT(/1X26HMESHING FREQUENCY IN CPS #,E13.5)                      00710000
              115   FORMAT(/7X2HFC11X2HFC11X2HEE)                                       00720000
              120   FORMAT(6E13.5)                                                      00730000
              125   FORMAT(/7X1HH12X2HTR11X4HTMAX5X2HSM)                                00740000
              130   FORMAT(/1X31HPOWER SPECTRAL DENSITY FUNCTION)                       00750000
              140   FORMAT(/6X,1HK6X2HFR11X2HGK11X3HGKK10X2HKM)                         00760000
              150   FORMAT(I7,3E13.5,F13.5)                                             00770000
                    RETURN                                                              00780000
                    END                                                                 00790000
```

```
FORTRAN IV G LEVEL  21                 TLU                   DATE = 75099            12/49/12

 0001                  SUBROUTINE TLU(A,B,C,D,N)                                              00010000
                 C**** MEMBER NAME R75TLU                                                     00020000
                 C              LINEAR INTERPOLATION ROUTINE                                  00030000
                 C     A* INDEPENDENT VARIABLE                                                00040000
                 C     B* DEPENDENT VARIABLE  $ANSWER<                                        00050000
                 C     C* INDEPENDENT TABLE                                                   00060000
                 C     D* DEPENDENT TABLE                                                     00070000
                 C     N* NUMBER OF POINTS IN TABLE                                           00080000
                 C     INDEPENDENT TABLE MUST BE SORTED, EITHER ASCENDING OR DESCENDING       00090000
 0002                  DIMENSION C(1),D(1)                                                    00100000
 0003                  IF(A.LT.C(1).OR.A.GT.C(N)) GO TO 1                                     00110000
 0004                  IF(N-1)1,2,3                                                           00120000
 0005                1 B=0.                                                                   00130000
 0006                  GO TO 100                                                              00140000
 0007                2 B=D(1)                                                                 00150000
 0008                  GO TO 100                                                              00160000
 0009                3 ML=1                                                                   00170000
 0010                  MU=N                                                                   00180000
 0011                8 IF(MU-ML-1) 15,15,9                                                    00190000
 0012                9 M=(MU+ML)/2                                                            00200000
 0013                  IF(C(1)-C(2))11,2,10                                                   00210000
 0014               10 IF(C(M)-A)13,12,14                                                     00220000
 0015               11 IF(A-C(M))13,12,14                                                     00230000
 0016               12 B=D(M)                                                                 00240000
 0017                  GO TO 100                                                              00250000
 0018               13 MU=M                                                                   00260000
 0019                  GO TO 8                                                                00270000
 0020               14 ML=M                                                                   00280000
 0021                  GO TO 8                                                                00290000
 0022               15 CONTINUE                                                               00300000
 0023                  B=D(ML)+(D(MU)-D(ML))*((A-C(ML))/(C(MU)-C(ML)))                        00310000
 0024              100 CONTINUE                                                               00320000
 0025                  RETURN                                                                 00330000
 0026                  END                                                                    00340000
```

APPENDIX C

LISTING OF COMPUTER PROGRAM FOR PREDICTION OF GEAR MESH EXCITATION SPECTRA – GGEAR (R-67)

RUN VERSION 2.3 --PSR LEVEL 332--

```
              PROGRAM GGEAR(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE22)            GGEAR    2
000003        COMMON/YY/UJC1(25),UJD1(25),UJ2(25),UJC2(25),UJD2(25),ZJ1(50)         CZIG     1
             1,ZJC1(25),ZJD1(25),ZJ2(50),ZJC2(25),ZJD2(25),QJ1(50),QJC1(25),QJD1   GGEAR    4
             2(25),QJ2(50),QJC2(25),            FJ1(50),UJ1(50),        CJC1(2     GGEAR    5
             35),CJD1(25),        AJ1(50),AJ2(50),QJC(50),QJD(50),ZJC(50),ZJD(50   GGEAR    6
             4),VJ(50),EJ1(56),FJ2(50),AJC1(25),AJC2(25),FJC1(27),QJD2(25),ZM2(    GGEAR    7
             550),UM2(50),WTC(27)                                                  GGEAR    8
000003        COMMON PJ(50),CJ(50),OJ(50),AJ(50),XJ(50),YJ(50),                     GGEAR    9
             1A(1500),B(1500),G(3000)                                              MAY11    1
000003        COMMON DO,BR,BA,FJ,CC,YM,NN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M          ALG      1
000003        COMMON QA(50),QB(50),QC(50),QA1(50),QB1(50),QC1(50),QA2(50),QB2(50   GGEAR   12
             1),QC2(50) ,IM,IP                                                     GGEAR   13
000003        COMMON WT(50),HEAD1(6),HEAD2(6)                                      GGEAR   14
000003        COMMON/GU/NMC,INT,MMM,IPLT,FN1,FN2,RB1,RO1,RO2,RT1,RT2,RM1,RM2,FI,   ALG      2
             1T1,T2,F1,F2,RF1,RF2,YE1,YE2,GE1,GE2,POS1,POS2,ANG                    ALG      3
000003        DIMENSION GGG(3000),WTCC(3000),FJCC1(3000),AANG(3000)                MAY11    2
000003        NR=5                                                                 GGEAR   18
000004          6                                                                  GGEAR   19
000005    198 RE  (NR,100) (HEAD1(I),I=1,4),(HEAD2(I),I=1,4)                       GGEAR   20
000025        REA (NR,117) NMC,INT,MN,MMM,IPLT,IFOUR,ISPECT                        MAY9     1
000047        READ(NR,112)FN1,FN2,RB1,RO1,RO2                                      GGEAR   22
000065        READ(NR,112)RT1,RT2,RM1,RM2,FI,T1                                    GGEAR   23
000105        READ(NR,112)T2,F1,F2,RF1,RF2                                         GGEAR   24
000123        READ(NR,112)YE1,YE2,GE1,GE2,POS1,POS2                                GGEAR   25
000143        WRITE(NW,100) (HEAD1(I),I=1,4),(HEAD2(I),I=1,4)                      GGEAR   26
000163        WRITE(NW,101)                                                        GGEAR   27
000167        WRITE(NW,108)                                                        GGEAR   28
000173        WRITE(NW,109) NMC,INT,MN,MMM                                         GGEAR   29
000207        WRITE(NW,104)                                                        GGEAR   30
000213        WRITE(NW,102) FI,FN1,RB1,RO1,RT1,RM1                                 GGEAR   31
000233        IF(MN) 511,510,511                                                   GGEAR   32
000236    510 WRITE(NW,103)                                                        GGEAR   33
000240        GO TO 512                                                            GGEAR   34
000241    511 WRITE(NW,160)                                                        GGEAR   35
000245    512 WRITE(NW,150)THETP,FN2,RO2,RT2,RM2                                   GGEAR   36
000263        WRITE(NW,105)                                                        GGEAR   37
000267        WRITE(NW,102)T1,F1,RF1,YE1,GE1,POS1                                  GGEAR   38
000307        WRITE(NW,107)                                                        GGEAR   39
000313        WRITE(NW,102)T2,F2,RF2,YE2,GE2,POS2                                  GGEAR   40
000333        WRITE(NW,164)                                                        GGEAR   41
000337        ANG=0.                                                               GGEAR   42
000340        IMC=1.                                                               GGEAR   43
000342        N=2.*FI+1.                                                           GGEAR   44
000345        NN=N*N                                                               GGEAR   45
000346        IF(MN.EQ.0) T2=-T2                                                   JUNE14   1
000351    150 WRITE(NW,1500)                                                       MAY3     2
000355        READ(NR,112)CL,VPT1,VPT2                                             ATT1     1
000367        WRITE(NW,1000) IMC,CL                                                GGEAR   47
000377        WRITE(NW,110)                                                        FINAL    1
000403        L=NN                                                                 CZIG     2
000405        DO 209 I=1,NN                                                        GGEAR   48
000406        READ(NR,112)ZJ1(I),UJ1(I),ZJ2(I),UJ2(I),WT(I)                        GGEAR   49
000423        WRITE(NW,102)ZJ1(I),UJ1(I),ZJ2(I),UJ2(I)                             GGEAR   50
000437        ZM2(L)=ZJ2(I)                                                        GGEAR   51
```

```
000442          UM2(L)=UJ2(I)                                          GGEAR    52
000444          IF(L-1) 209,209,225                                    GGEAR    53
000446      225 L=L-1                                                  GGEAR    54
000450      209 CONTINUE                                               GGEAR    55
000453          CALL GEARO(CL,VPT1,VPT2)                               GGEAR    56
000455          NNN=IMC*N                                              GGEAR    57
000460          DO 200 I=1,N                                           GGEAR    58
000461          JJ=NNN+I-N                                             GU        1
000463          AANG(JJ)=AJC1(I)                                       MAY3      3
000466          GGG(JJ)=EJ1(I)                                         GGEAR    60
000470          G(JJ)=GGG(JJ)                                          GGEAR    61
000472          WTCC(JJ)=WTC(I)                                        GGEAR    62
000474      200 FJCC1(JJ)=JJ                                           MAY2      1
000500          IMC=IMC+1                                              GGEAR    64
000501          IF(IMC.LE.NMC) GO TO 150                               GGEAR    65
000503          WRITE(NW,1500)                                         MAY3      4
000507          WRITE(NW,2000)                                         MAY3      5
000513          DO 2500 I=1,NNN                                        MAY3      6
000515     2500 WRITE(NW,102)FJCC1(I),AANG(I),GGG(I)                   MAY3      7
000531          IF(IPLT.EQ.1) CALL PLT(FJCC1,GGG,WTCC,NNN)             GGEAR    66
000536          M=FI                                                   GU        3
000540          IF(IFOUR.NE.1) GO TO 3000                              MAY9      2
000542          WRITE(NW,128)                                          ALG       4
000546          CALL FOUR                                              GGEAR    67
000547          LL=MMM+1                                               GGEAR    68
000551          DO 250 I=1,LL                                          GGEAR    69
000553          LP=I-1                                                 GGEAR    70
000554          SMK=1.*LP/NMC                                          JUNE7     1
000560          RESLT=SQRT(A(I)**2+B(I)**2)                            GGEAR    71
000566      250 WRITE(NW,129)LP,A(I),B(I),RESLT,SMK                    JUNE7     2
000606     3000 IF(ISPECT.NE.1) GO TO 3500                             MAY9      3
000610          CALL SPECT                                             MAY9      4
000611     3500 CONTINUE                                               MAY9      5
000611          IF(INT)300,190,300                                     GGEAR    73
000612      300 CALL EXIT                                              GGEAR    74
000613      190 FORMAT(8A10)                                           GGEAR    75
000613      101 FORMAT(72H0   PN0335=COMPUTATIONS OF GEAR TOOTH MESHING ERRORS   GGEAR    76
               1  3-22-1966       )                                    GGEAR    77
000613      102 FORMAT(7E13.5)                                         MAY9      6
000613      103 FORMAT(12H0   PRE.ANGLE7X2HN224X3HRI210X3HRT210X3HRM2)  GGEAR    79
000613      104 FORMAT(/7X1HT11X2HN111X3HRB110X3HRO110X3HRT110X3HRM1)  GGEAR    80
000613      105 FORMAT(/6X2HT111X2HF111X3HRF15X12HYOUNGS MOD-12X11HSHEAR MOD-12X11 GGEAR    81
               1HPOS.RATIO-1)                                          GGEAR    82
000613      107 FORMAT(/6X2HT211X2HF211X3HRF25X12HYOUNGS MOD-22X11HSHEAR MOD-22X11 GGEAR    83
               1HPOS.RATIO-2)                                          GGEAR    84
000613      108 FORMAT(50H0 MESH CYCLES     INPUT,    DIFF.GEAR N.OF HARMONIC)  GGEAR    85
000613      109 FORMAT(4X14,4(8X14))                                   GGEAR    86
000613      110 FORMAT(60H0INPUT LISTING OF PROFILE ERROR AND SUPPLEMENTARY COMPLI FINAL     2
               1ANCE/6X2HZ111X2HU111X2HZ211X2HU2)                      FINAL     3
000613      112 FORMAT(6E13.5)                                         GGEAR    87
000613      117 FORMAT(8I5)                                            MAY9      7
000613      128 FORMAT(43H0CALCULATED FOURIER COEFFICIENTS FOR ERRORS//7H      19X ALG       5
               15HA(I) 13X4HB(I)14X1HC14X2HKM)                         JUNE7     3
000613      129 FORMAT(I7,3(4X  E14.7),F14.7)                          JUNE7     4
000613      158 FORMAT(2E13.5,13X,3E13.5)                              GGEAR    90

000613      160 FORMAT(12H0          7X2HN224X3HRO210X3HRT210X3HRM2)   GGEAR    91
000613      164 FORMAT(//16H CALCULATED DATA)                          GGEAR    92
000613     1000 FORMAT(16H MESH CYCLE NO.=,I3,20H,   CENTER DISTANCE=,E13.6)  MAY14     1
000613     1500 FORMAT(/6X 20(H*3X)/)                                  MAY3      8
000613     2000 FORMAT(5X 3HJC19X4HAJC16X11HTANG. ERROR)               MAY3      9
000613          STOP                                                   GGEAR    94
000615          END                                                    GGEAR    95
```

```
RUN VERSION 2.3 --PSR LEVEL 332--

                 SUBROUTINE GEARO(CL,VPT1,VPT2)                                       JUN13      1
000006           COMMON/XY/UJC1(25),UJ01(25),UJ2(25),UJC2(25),UJD2(25),ZJ1(50)       CZIG       3
                1,ZJC1(25),ZJ01(25),ZJ2(50),ZJC2(25),ZJD2(25),QJ1(50),QJC1(25),QJD1 GEARO      4
                2(25),QJ2(50),QJC2(25),             FJ1(50),UJ1(50),        CJC1(2 GEARO      5
                35),CJ01(25),          AJ1(50),AJ2(50),QJC(50),QJD(50),ZJC(50),ZJD(50 GEARO   6
                4),VJ(50),EJ1(56),FJ2(50),AJC1(25),AJC2(25),FJC1(27),QJD2(25),ZM2(  GEARO      7
                550),UM2(50),WTC(27)                                                 GEARO      8
000006           COMMON PJ(50),CJ(50),QJ(50),AJ(50),XJ(50),YJ(50),                   GEARO      9
                1A(1500),B(1500),G(3000)                                             MAY11      3
000006           COMMON DD,BR,BA,FJ,CC,YM,NN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M        ALG        7
000006           COMMON QA(50),QB(50),QC(50),QA1(50),QB1(50),QC1(50),QA2(50),QB2(50 GEARO     12
                1),QC2(50) ,IM,IP                                                    GEARO     13
000006           COMMON WT(50),HEAD1(6),HEAD2(6)                                     GEARO     14
000006           COMMON/GU/NMC,INT,MMM,IPLT,FN1,FN2,RB1,RO1,RO2,RT1,RT2,RM1,RM2,FI,  ALG        8
                1T1,T2,F1,F2,RF1,RF2,YE1,YE2,GE1,GE2,POS1,POS2,ANG                   ALG        9
           C SPECIFY AND INITIALIZE   READING AND WRITING UNITS FOR IBM 1800         GEARO     17
000006           NR=5                                                                GEARO     18
000006           NW=6                                                                GEARO     19
000010       198 CONTINUE                                                            GEARO     20
000010           RP1=CL*FN1/(FN1+FN2)                                                APR30      1
000013           IF(MN.EQ.0)RP1=CL*FN1/(FN2-FN1)                                     JUNE7      5
000017           CTT=RB1/RP1                                                         GEARO     22
000021           TTT=SQRT(1.-CTT**2)/CTT                                             GEARO     23
000026           THETP=ATAN(TTT)                                                     JUNE5      1
000030           THETPP=THETP*180./3.1415927                                         JUNE5      2
000033           WRITE(NW,112)                                                       YUNG       1
000036           WRITE(NW,102) THETPP                                                JUNE5      3
000044           IF(F1-F2) 320,320,321                                               GEARO     25
000051       321 FF2=F2                                                              GEARO     26
000053           GO TO 322                                                           GEARO     27
000053       320 FF2=F1                                                              GEARO     28
000055       322 FF2=FF2**0.8                                                        GEARO     29
000061       304 DD1=-T1/2.0/RP1+THETP                                               GEARO     33
000065           F12=FN1/FN2                                                         GEARO     34
000067           RP2=RP1/F12                                                         GEARO     35
000070           DD2=-T2/2.0/RP2+THETP                                               GEARO     36
000074           FNJ=2.0*FI+1.0                                                      GEARO     37
000077           N=FNJ                                                               GEARO     38
000101           NN=N+N                                                              GEARO     39
000102           CS=CTT                                                              JUNE6      1
000104           TAN=TTT                                                             JUNE5      4
000105           EP1=1.0-POS1*POS1                                                   GEARO     43
000107           EP1=YE1/EP1                                                         GEARO     44
000111           EP2=1.0-POS2*POS2                                                   GEARO     45
000114           EP2=YE2/EP2                                                         GEARO     46
000115           EP12=(1.0/EP1+1.0/EP2)*0.5                                          GEARO     47
000121           EP12=EP12**0.9                                                      GEARO     48
000125           RB2=RB1/F12                                                         JUNE5      5
000127           C1=RO1/RB1                                                          GEARO     51
000130           C2=RO2/RB2                                                          GEARO     52
000132           BR2=(SQRT (C2*C2-1.0)-TAN)                                          GEARO     53
000140           BR1=(SQRT (C1*C1-1.0)-TAN)                                          GEARO     54
000146           C3=TAN/F12                                                          GEARO     55
000147           IF(BR1-C3) 403,403,404                                              GEARO     56
```

```
000154      404 BR1=C3                                        GEARO     57
000156          WRITE(NW,126)BR1                              GEARO     58
000163      403 BA1=BR2/F12                                   GEARO     59
000165          BA2=BR1*F12                                   GEARO     60
000166          IF(MN) 440,450,440                            GEARO     61
000172      450 BA1=-BA1                                      GEARO     62
000173          BA2=-BA2                                      GEARO     63
000175      440 IF(BA1-TAN) 401,401,402                       GEARO     64
000200      402 BA1=TAN                                       GEARO     65
000202          WRITE(NW,125)BA1                              GEARO     66
000207      401 C1=6.2831845/FN1                              GEARO     67
000211          IF(BA1-C1) 213,213,611                        GEARO     68
000216      611 WRITE(NW,174)                                 GEARO     69
000222      213 IF(BR1-C1) 215,612,612                        GEARO     70
000227      612 WRITE(NW,175)                                 GEARO     71
000233      215 C1=RT1/RB1                                    GEARO     72
000235          C2=RT2/RB2                                    GEARO     73
000237          C1 =SQRT (C1*C1-1.0)-TAN                      GEARO     74
000245          AT1=ABS (C1)                                  GEARO     75
000246          C2 =SQRT (C2*C2-1.0)-TAN                      GEARO     76
000255          AT2=ABS (C2)                                  GEARO     77
000257          WRITE(NW,122)                                 GEARO     78
000262          WRITE(NW,102)RP1,RB1,BA1,BR1,AT1              FINAL      6
000300          WRITE(NW,162)                                 GEARO     80
000304          WRITE(NW,102)RP2,RB2,BA2,BR2,AT2              GEARO     81
000322          NMZ=1                                         JUN13      2
000323          NUZ=1                                         JUN13      3
000324          IF(AT1-BA1) 311,216,216                       GEARO     84
000331      311 WRITE(NW,176)                                 GEARO     85
000335      216 IF(AT2-BA2) 309,217,217                       GEARO     86
000342      309 IF(MN) 620,621,620                            GEARO     87
000343      621 WRITE(NW,178)                                 GEARO     88
000347          GO TO 217                                     GEARO     89
000352      620 WRITE(NW,179)                                 GEARO     90
000356      217 IF(RM1-RB1) 191,192,192                       GEARO     91
000363      191 AM1=1.0                                       GEARO     92
000365          WRITE(NW,118)                                 GEARO     93
000370          GO TO 1920                                    GEARO     94
000373      192 AM1=RM1/RB1                                   GEARO     95
000375     1920 C1=AM1                                        GEARO     96
000376          AM1=SQRT (AM1*AM1-1.0)-TAN                    GEARO     97
000404          C2=RF1/RB1                                    GEARO     98
000406          PF1=C2                                        GEARO     99
000407          C1=RM1/RB1                                    GEARO    100
000411          C3=(C1+C2)**2-1.0                             GEARO    101
000414          IF(C3) 406,406,407                            GEARO    102
000420      406 C3=0.0                                        GEARO    103
000421          WRITE(NW,156)                                 GEARO    104
000425          TPO1=0.0                                      GEARO    105
000426          WRITE(NW,156)PO1                              GEARO    106
000434          GO TO 408                                     GEARO    107
000437      407 PO1=SQRT (C3)                                 GEARO    108
000441          TPO1=ATAN (PO1)                               GEARO    109
000444      408 IF(RM2-RB2) 193,194,194                       GEARO    110
000451      193 AM2=1.0                                       GEARO    111
```

```
000453            WRITE(NW,119)
000456            GO TO 1940
000461       194  AM2=RH2/RB2
000463      1940  C1=AM2
000464            AM2=SQRT (AM2*AM2-1.0)-TAN
000472            C2=RF2/RB2
000474            PF2=C2
000475            C1=RM2/RB2
000477            IF(MN) 456,455,456
000502       455  C2=-C2
000503       456  C3=(C1+C2)**2-1.0
000506            IF(C3) 409,409,410
000510       409  C3=0.0
000511            WRITE(NW,155)
000515            TPD2=0.0
000516            WRITE(NW,155)PD2
000524            GO TO 411
000527       410  PD2=SQRT (C3)
000531            TPD2=ATAN (PD2)
000534       411  K1=1
000535            E1=PF1
000536            E2=PD1
000540            E3=TPD1
000541            E4=RM1
000543            DD=DD1
000544            C1=AM1
000546       201  C2=DD+C1
000550            C1=ATAN (TAN+C1)
000554            C4=C1-C2
000556            C3=E3-E2-DD*TAN
000562            IF(K1-1) 422,422,425
000567       425  IF(MN) 422,421,422
000570       421  C3=-C3
000571            C4=-C4
000573       422  C1=C3+E1
000575            C4=(C1+C4)*0.5
000600            Y=E4*SIN (C4)
000602            X=E4*COS (C4)
000605            IF(K1-1) 202,202,203
000612       202  XM1=X
000613            YM1=Y
000615            K1=K1+1
000617            DD=DD2
000620            C1=AM2
000621            E1=PF2
000623            E2=PD2
000624            E3=TPD2
000626            E4=RM2
000630            GO TO 201
000630       203  XM2=X
000631            YM2=Y
000633            EE=6.2831854/FNJ
000635            CC=EE/FN1
000636            FJ =-FNJ+1.0
000640            RB=RB1
```

```
000641          BR=BR1
000643          BA=BA1
000644          DD=DD1
000646          XM=XM1
000647          YM=YM1
000651          GM=GE1
000652          F=F1
000654          II=1
000655          EP=EP1
000657          WRITE(NW,151)
000662          CALL AJCDH
000663          DO 221 I=1,NN
000667          C1     =XJ(I)
000670          C2     =YJ(I)
000672          FJ1(I)=PJ(I)
000674          C3=CJ(I)
000675          AJ1(I)=AJ(I)
000677          QJ1(I)=QJ(I)
000701          IH1=IH
000702          IP1=IP
000704          QA1(I)=QA(I)
000706          QB1(I)=QB(I)
000710          QC1(I)=QC(I)
000712          IF(C3) 515,515,516
000714      515 WRITE(NW,102)FJ1(I),C3,AJ1(I)
000726          GO TO 221
000731      516 WRITE(NW,102)FJ1(I),C3,AJ1(I),QJ1(I),C1,C2
000751      221 CONTINUE
000756          WRITE(NW,120)XM1,YM1
000765          WRITE(NW,161)
000771          DO 517 I=IH1,IP1
000775      517 WRITE(NW,102)FJ1(I),QA1(I),QB1(I),QC1(I)
001015          DD=DD2
001016          RB=RB2
001017          XM=XM2
001021          YM=YM2
001022          IF(MN) 302,301,302
001024      301 FJ=FNJ
001026          GO TO 310
001026      302 FJ=-FNJ
001030      310 F=F2
001031          EP=EP2
001033          GM=GE2
001034          CC=EE/FN2
001036          II=2
001040          WRITE(NW,152)
001043          CALL AJCDH
001044          L=NN
001046          DO 222 I=1,NN
001051          C4     =XJ(I)
001052          C5     =YJ(I)
001054          IH2=IH
001055          IP2=IP
001057          QA2(I)=QA(I)
001061          QB2(I)=QB(I)
```

```
001063          QC2(I)=QC(I)                                        GEARO   222
001065          FJ2(I)=PJ(I)                                        GEARO   223
001067          C1=AJ(I)                                            GEARO   224
001070          AJ2(L)=C1                                           GEARO   225
001073          C2=QJ(I)                                            GEARO   226
001074          QJ2(L)=C2                                           GEARO   227
001076          IF(C2) 519,518,519                                  GEARO   228
001076      518 WRITE(NW,132)FJ2(I),C1                              GEARO   229
001106          GO TO 520                                           GEARO   230
001111      519 WRITE(NW,132)FJ2(I),C1,C2,C4,C5                     GEARO   231
001127      520 IF(L-1) 222,222,226                                 GEARO   232
001133      226 L=L-1                                               GEARO   233
001135      222 CONTINUE                                            GEARO   234
001140          WRITE(NW,120)XM2,YM2                                GEARO   235
001147          WRITE(NW,163)                                       GEARO   236
001153          DO 521 I=IM2,IP2                                    GEARO   237
001157      521 WRITE(NW,102)FJ2(I),QA2(I),QB2(I),QC2(I)            GEARO   238
001177          DO 251 I=1,N                                        GEARO   239
001200          IN=I+N                                              GEARO   240
001201          FJC1(I)=FJ1(IN)                                     GEARO   241
001204          AJC1(I)=AJ1(IN)                                     GEARO   242
001206          AJC2(I)=AJ2(IN)                                     GEARO   243
001210          QJC1(I)=QJ1(I)                                      GEARO   244
001212          QJC1(I)=QJ1(IN)                                     GEARO   245
001214          QJC2(I)=QJ2(IN)                                     GEARO   246
001216          QJD2(I)=QJ2(I)                                      GEARO   247
001220          CJD1(I)=CJ (I)                                      GEARO   248
001222      251 CJC1(I)=CJ (IN)                                     GEARO   249
001226          MUZ=1                                               GEARO   250
001227          MMZ=1                                               GEARO   251
001230      197 CONTINUE                                            MAY1      1
001230          L=NN                                                GEARO   253
001232          DO 415 I=1,N                                        GEARO   254
001233          IN=I+N                                              GEARO   255
001234          UJD1(I)=UJ1(I)                                      GEARO   256
001236          UJC1(I)=UJ1(IN)                                     GEARO   257
001241          UJC2(I)=UM2(IN)                                     GEARO   258
001243          UJD2(I)=UM2(I)                                      GEARO   259
001245          ZJD1(I)=ZJ1(I)                                      GEARO   260
001247          ZJC1(I)=ZJ1(IN)                                     GEARO   261
001251          ZJC2(I)=ZM2(IN)                                     GEARO   262
001253          ZJD2(I)=ZM2(I)                                      GEARO   263
001255          QJC(I)=QJC1(I)+UJC1(I)+QJC2(I)+UJC2(I)              GEARO   264
001263          QJD(I)=QJD1(I)+UJD1(I)+QJD2(I)+UJD2(I)              GEARO   265
001273      415 ZJC(I)=ZJC1(I)+ZJC2(I)                              GEARO   266
001301      199 CONTINUE                                            GEARO   267
001301          WRITE(NW,106)                                       GEARO   268
001305          WRITE(NW,102)VPT1,VPT2                              GEARO   269
001317          VP1=VPT1/CS                                         GEARO   270
001322          VP2=VPT2/CS                                         GEARO   271
001324      418 CONTINUE                                            MAY1      2
001324          WRITE(NW,115)                                       GEARO   275
001330          DO 501 I=1,N                                        GEARO   276
001334          WT(I)=WT(N+I)                                       GEARO   277
001337          WN=WT(1)/CS                                         GEARO   278
```

```
001341          IF(WN)214,214,800                                              GEARO     279
001343      800 CONTINUE                                                       GEARO     280
001343          C1=WN**.1                                                      MAY1        3
001347          QD=1.37/FF2*EP12/C1                                            MAY1        4
001352          ZJD(I)=ZJD1(I)+ZJD2(I)+VP1-VP2                                 GEARO     281
001360          CCC    =QJC(I)+QD                                              GEARO     282
001362          DDD    =QJD(I)+QD                                              GEARO     283
001364          C1=CJC1(I)                                                     GEARO     284
001366          C2=CJD1(I)                                                     GEARO     285
001367          IF(C1) 256,256,257                                             GEARO     286
001371      256 IF(C2) 501,501,258                                             GEARO     287
001373      257 IF(C2) 259,259,260                                             GEARO     288
001375      259 VJ(I)=ZJC(I)-CCC    *WN                                        GEARO     289
001401          WTC(I)=WN*CS                                                   GEARO     290
001403          WTD=0.0                                                        GEARO     291
001404          GO TO 261                                                      GEARO     292
001405      258 VJ(I)=ZJD(I)-DDD    *WN                                        GEARO     293
001411          WTC(I)=0.0                                                     GEARO     294
001412          WTD=WN*CS                                                      GEARO     295
001414          GO TO 261                                                      GEARO     296
001415      260 C3=CCC    *WN+ZJD(I)-ZJC(I)                                    GEARO     297
001422          C4=DDD    *WN+ZJC(I)-ZJD(I)                                    GEARO     298
001426          IF(C3) 259,259,262                                             GEARO     299
001430      262 IF(C4) 258,258,263                                             GEARO     300
001432      263 C1=CCC+DDD                                                     GEARO     301
001434          C2=CCC    *ZJD(I)+DDD    *ZJC(I)-CCC*DDD*WN                    GEARO     302
001442          VJ(I)=C2/C1                                                    GEARO     303
001444          C1=CS/C1                                                       GEARO     304
001445          WTC(I)=(DDD*WN+ZJC(I)-ZJD(I))*C1                               GEARO     305
001452          WTD=(CCC*WN+ZJD(I)-ZJC(I))*C1                                  GEARO     306
001460      261 EJ1(I)=VJ(I)/CS                                                GEARO     307
001463          AJC1(I)=ANG+AJC1(I)                                            GEARO     308
001465          WRITE(NW,102)FJC1(I),AJC1(I),EJ1(I),WTC(I),WTD,WN,WT(I),QD     MAY1        5
001511      501 CONTINUE                                                       GEARO     310
001516          ANG=AJC1(N)                                                    GEARO     311
001517       24 IF(NUZ-MUZ) 504,504,502                                        GEARO     312
001522      502 MUZ=MUZ+1                                                      GEARO     313
001524          GO TO 199                                                      GEARO     314
001524      504 MUZ=1                                                          GEARO     315
001525          IF(NMZ-MMZ) 214,214,506                                        GEARO     316
001530      506 MMZ=MMZ+1                                                      GEARO     317
001532          GO TO 197                                                      GEARO     318
001532      214 CONTINUE                                                       GEARO     319
001532      503 RETURN                                                         GEARO     320
001533      102 FORMAT(8E13.5)                                                 MAY1        6
001533      106 FORMAT(//34H INPUT DATA ON TOOTH SPACING ERROR/6X4HVPT19X4HVPT2)   GEARO     322
001533      110 FORMAT(60H0INPUT LISTING OF PROFILE ERROR AND SUPPLEMENTARY COMPLI GEARO     323
               1ANCE/6X2HZ111X2HU111X2HZ211X2HU2)                              GEARO     324
001533      112 FORMAT(16H0PRES.ANGLE(DEG))                                    FINAL       7
001533      115 FORMAT(42H0CALCULATED TOOTH MESHING ERRORS AND LOADS//6X3HJC19X4HA GEARO     325
               1JC16X11HTANG. ERROR6X3HWTC10X3HWTD11X2HWN11X2HWT10X3HQJD)      MAY1        7
001533      118 FORMAT(74H0DRIVING GEAR INPUT ROOT RADIUS SMALLER THAN BASE CIRCLE GEARO     327
               1 RADIUS                        )                               GEARO     328
001533      119 FORMAT(74H0DRIVEN  GEAR INPUT ROOT RADIUS SMALLER THAN BASE CIRCLE GEARO     329
               1 RADIUS                        )                               GEARO     330
```

```
001533       120 FORMAT(48H0COOD. OF EFFECTIVE TOOTH PROFILE AT ROOT CIRCLE11X1HX13 GEARO     331
                 1X1HY/52X,2E13.6)                                                 GEARO     332
001533       122 FORMAT(/6X3HRP110X3HRB110X3HBA110X3HBR110X3HAT1)                  ALG        10
001533       125 FORMAT(66H0 DRIVEN GEAR TEETH ENGAGE UNDER CUT PORTION OF DRIVING GEARO     334
                 1GEAR TEETH//6H  BA1=, E13.6)                                     GEARO     335
001533       126 FORMAT(66H0 DRIVING GEAR TEETH ENGAGE UNDER CUT PORTION OF DRIVEN GEARO     336
                 1GEAR TEETH//6H  BR1=, E13.6)                                     GEARO     337
001533       132 FORMAT(  E13.5,13X,4E13.5)                                        GEARO     338
001533       151 FORMAT(/6X2HJ111X3HCJ110X3HAJ18X6HQJ1A8C9X3HXJ110X3HYJ1)          GEARO     339
001533       152 FORMAT(/6X2HJ224X3HAJ28X6HQJ2A8C9X3HXJ210X3HYJ2)                  GEARO     340
001533       155 FORMAT(63H0DRIVEN  GEAR INPUT RADIUS TO FILLET CENTER INSIDE BASE GEARO     341
                 1CIRCLE./42H0PROGRAM CONTINUES WITH CORRECT TREATMENT.)           GEARO     342
001533       156 FORMAT(63H0DRIVING GEAR INPUT RADIUS TO FILLET CENTER INSIDE BASE GEARO     343
                 1CIRCLE./42H0PROGRAM CONTINUES WITH CORRECT TREATMENT.)           GEARO     344
001533       159 FORMAT(13X,4E13.5)                                                GEARO     345
001533       161 FORMAT(/6X2HJ110X4HQJ1A9X4HQJ1B9X4HQJ1C)                          GEARO     346
001533       162 FORMAT(/6X3HRP210X3HRB210X3HBA210X3HBR210X3HAT2)                  GEARO     347
001533       163 FORMAT(/6X2HJ210X4HQJ2A9X4HQJ2B9X4HQJ2C)                          GEARO     348
001533       170 FORMAT(43H0CALCULATED TOTAL CONTACT COMPLIANCE   QJD=,E13.6)      GEARO     349
001533       171 FORMAT(23H0CALCULATED NORMAL  WN=, E13.6)                         GEARO     350
001533       174 FORMAT(80H0 ANGLE OF APPROACH ON DRIVING GEAR IS GREATER THAN TOOT GEARO    351
                 1H SPACING ANGLE. PROGRAM/25HCONTINUED WITHOUT OVERLAP)           GEARO     352
001533       175 FORMAT(75H0 ANGLE OF RECESS ON DRIVING GEAR IS NOT SMALLER THAN TO GEARO    353
                 1OTH SPACING ANGLE. /34H PROGRAM CONTINUED WITHOUT OVERLAP)       GEARO     354
001533       176 FORMAT(80H0 DRIVING GEAR TEETH MESHING ON PROFILE INSIDE OF TIF DI GEARO    355
                 1AMETER. PROGRAM CONTINU-/21HED WITHOUT CORRECTION)               GEARO     356
001533       178 FORMAT(80H0 DRIVEN  GEAR TEETH MESHING ON PROFILE OUTSIDE OF TIF D GEARO    357
                 1IAMETER. PROGRAM CONTIN-/22HUED WITHOUT CORRECTION)              GEARO     358
001533       179 FORMAT(80H0 DRIVEN  GEAR TEETH MESHING ON PROFILE INSIDE OF TIF DI GEARO    359
                 1AMETER. PROGRAM CONTINU-/21HED WITHOUT CORRECTION)               GEARO     360
001533           END                                                               GEARO     361
```

RUN VERSION 2.3 --PSR LEVEL 332--

```
                    SUBROUTINE AJCOH                                               AJCOH
000002              DIMENSION SD(50),CD(50),BIK(50),AIK(50),FLK(50) ,CDC(50)       AJCOH
000002              COMMON PJ(50),CJ(50),QJ(50),AJ(50),XJ(50),YJ(50),              MAY11
                   1A(1500),B(1500),G(3000)                                        MAY11
000002              COMMON DD,BR,BA,FJ,CC,YM,NN,N,FNJ,EP,TAN,RB,GH,F,XM,MN,II,M    AJCOH
000002              COMMON QA(50),QB(50),QC(50),QA1(50),QB1(50),QC1(50),QA2(50),QB2(50 AJCOH
                   1),QC2(50),IH,IP                                                AJCOH
000002              DO 106 I=1,NN                                                  AJCOH
000004              QA(I)=0.0                                                      AJCOH
000005              QB(I)=0.0                                                      AJCOH
000006              QC(I)=0.0                                                      AJCOH
000007              QJ(I)=0.0                                                      AJCOH
000010              XJ(I)=0.0                                                      AJCOH
000011          106 YJ(I)=0.0                                                      AJCOH
000014              IF(II-1) 405,405,403                                           AJCOH
000017          405 MM=1.0                                                         AJCOH
000021              GO TO 404                                                      AJCOH
000022          403 MM=MN                                                          AJCOH
000024          404 K=1                                                            AJCOH
000025              DO 501 I=1,NN                                                  AJCOH
000027              AJ(I)=FJ*CC                                                    AJCOH
000031              PJ(I)=FJ                                                       AJCOH
000032              IF(II-1) 412,412,413                                           AJCOH
000035          412 CALL CALCJ(AJ(I),BR,BA,XY)                                     AJCOH
000042              CJ(I)=XY                                                       AJCOH
000044              GO TO 414                                                      AJCOH
000045          413 XY=CDC(I)                                                      AJCOH
000047          414 IF(XY) 503,503,101                                             AJCOH
000051          101 IF(K-1) 102,102,103                                            AJCOH
000054          102 IH=I                                                           AJCOH
000056          103 AX=AJ(I)                                                       AJCOH
000060              DJ=DD+AX                                                       AJCOH
000062              SD(I)=SIN (DJ)                                                 AJCOH
000066              CD(I)=COS (DJ)                                                 AJCOH
000072              C1=ATAN (TAN*AX)                                               AJCOH
000076              C3=COS (C1)                                                    AJCOH
000100              C4=C1-DJ                                                       AJCOH
000102              IF(MM) 402,401,402                                             AJCOH
000104          401 C4=-C4                                                         AJCOH
000105          402 C5=SIN (C4)                                                    AJCOH
000107              C6=COS (C4)                                                    AJCOH
000111              C1=RB/C3                                                       AJCOH
000113              XJ(I)=C1*C6                                                    AJCOH
000116              YJ(I)=C1*C5                                                    AJCOH
000120              IF(K-1) 242,242,243                                            AJCOH
000122          242 BIK(I)=(YJ(I)**3+YM**3)/3.0*F                                  AJCOH
000130              AIK(I)=(YJ(I)+YM)*F                                            AJCOH
000133              FLK(I)=XJ(I)-XM                                                AJCOH
000136              GO TO 504                                                      AJCOH
000136          243 BIK(I)=(YJ(I)**3+YJ(I-1)**3)/3.0*F                             AJCOH
000144              AIK(I)=(YJ(I)+YJ(I-1))*F                                       AJCOH
000147              FLK(I)=XJ(I)-XJ(I-1)                                           AJCOH
000152          504 K=K+1                                                          AJCOH
000154          503 IF(I-NN) 502,501,501                                           AJCOH
```

```
000157      502 IF(MM) 506,505,506
000160      505 FJ=FJ-1.
000162          GO TO 501
000163      506 FJ=FJ+1.0
000165      501 CONTINUE
000170          IF(II-1) 301,301,302
000172      301 IF(MM) 601,302,601
000173      601 LL=NN
000175          DO 303 I=1,NN
000176          COC(I)=CJ(LL)
000200          IF(LL-1) 303,303,305
000203      305 LL=LL-1
000205      303 CONTINUE
000210      302 IP=K+IH-2
000213      410 DO 246 L=IH,IP
000215          E1=0.0
000216          E2=0.0
000216          E3=0.0
000217          E4=0.0
000220          EE=XJ(L)-XM
000223          IM=L
000223          IF(MM) 202,203,202
000225      203 EE=-EE
000226      202 DO 245 I=IH,IM
000230          C1      =XJ(L)-XJ(I)
000233          C5=FLK(I)
000234          IF(MM) 207,208,207
000236      208 C1=-C1
000237          C5=-C5
000241      207 C2=C5/BIK(I)
000243          C3=(C5+3.0*C1)*C5+3.0*C1*C1
000251          E4=E4+C2
000253          C4=2.0*C1+C5
000255          E1=E1+C3*C2
000260          E2=E2+C4*C2
000263      245 E3=E3+C5/AIK(I)
000271          IF(MM) 206,205,206
000272      205 IM=IM-1
000274      206 C1=CD(L)*CD(L)
000276          C2=SD(L)*YJ(L)
000300          QA(L) =C1/3.0*E1/EP+(C2*E4-CD(L)*E2)/EP*C2
000312          QB(L) =C1/GM*E3*1.2
000317          C1=(EE*CD(L)-YJ(L)*SD(L))/YM
000323          QC(L)=C1/EP*1.327/F*C1
000330      246 QJ(L)=QA(L)+QB(L)+QC(L)
000336          RETURN
000337          END
```

RUN VERSION 2.3 --PSR LEVEL 332--

```
                  SUBROUTINE FOUR                                                  FOUR      2
              C   REQUIRES 2*N+1 POINTS  F(I)                                      FOUR      3
              C   POINTS CORRESPOND TO THETA=2*PI/(2*N+1),...,2*PI                 FOUR      4
              C   OUTPUT  A(I),B(I)  REFER TO COSINE AND SINE OF (I-1)*THETA       FOUR      5
000002            COMMON PJ(50),CJ(50),GJ(50),AJ(50),XJ(50),YJ(50),                FOUR      6
                 1A(1500),B(1500),G(3000)                                          MAY11     6
000002            COMMON DO,BR,BA,FJ,CC,YM,NN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M      FOUR      8
000002            COMMON/GU/NMC,INT,MMM,IPLT,FN1,FN2,RB1,RO1,RO2,RT1,RT2,RM1,RM2,FI, MAY9   8
                 1T1,T2,F1,F2,RF1,RF2,YE1,YE2,GE1,GE2,POS1,POS2,ANG                MAY9      9
000002            C=1.0                                                            FOUR      9
000003            S=0.0                                                            FOUR     10
000004            M=IFIX(FI)*NMC+NMC/2                                             MAY9     10
000010            N1=M+1                                                           FOUR     11
000012            NA=N*NMC                                                         MAY9     11
000014            N2=NA-1                                                          MAY9     12
000015            T1=2./NA                                                         MAY9     13
000020            T2=T1*3.1415927                                                  FOUR     15
000021            C1=COS (T2)                                                      FOUR     16
000023            S1=SIN (T2)                                                      FOUR     17
000026            DO 7 IP=1,N1                                                     FOUR     18
000027            U1=0.0                                                           FOUR     19
000030            U2=0.0                                                           FOUR     20
000031            DO 3 I=1,N2                                                      FOUR     21
000032            J=N2-I+2                                                         FOUR     22
000034            U3=G(J)+2.0*C*U1-U2                                              FOUR     23
000042            U2=U1                                                            FOUR     24
000043          3 U1=U3                                                            FOUR     25
000047            A(IP)=T1*(G(1)+C*U1-U2)                                          FOUR     26
000054            B(IP)=T1*S*U1                                                    FOUR     27
000056            AA=A(IP)*C-B(IP)*S                                               FOUR     28
000062            BB=A(IP)*S+B(IP)*C                                               FOUR     29
000065            A(IP)=AA                                                         FOUR     30
000067            B(IP)=BB                                                         FOUR     31
000071            Q=C1*C-S1*S                                                      FOUR     32
000074            S=C1*S+S1*C                                                      FOUR     33
000077          7 C=Q                                                              FOUR     34
000103            RETURN                                                           FOUR     35
000103            END                                                              FOUR     36

                  SUBROUTINE CALCJ(C1,C2,C3,C4)                                    CALCJ     2
000007            IF(C3) 240,231,241                                               CALCJ     3
000011            DATA EHEAD3/4H LB./                                              CALCJ     4
000011        241 IF(C1) 230,218,218                                               CALCJ     5
000013        230 IF(ABS (C1)-C3)  220,220,219                                     CALCJ     6
000017        231 IF(C1) 219,218,218                                               CALCJ     7
000021        240 IF(C1) 219,219,242                                               CALCJ     8
000023        242 C3=-C3                                                           CALCJ     9
000024            IF(C1-C3) 219,218,218                                            CALCJ    10
000026        218 IF(C2-C1) 219,220,220                                            CALCJ    11
000030        220 C4=1.0                                                           CALCJ    12
000031            GO TO 221                                                        CALCJ    13
000032        219 C4=0.0                                                           CALCJ    14
000033        221 RETURN                                                           CALCJ    15
000034            END                                                              CALCJ    16
```

```
RUN VERSION 2.3 --PSR LEVEL 332--

                 SUBROUTINE PLT(X,Y,Z,NPTS)                                      PLT       2
000007           DIMENSION Y(3000),X(3000),Z(3000),AXX(4),AYY(4),AZZ(4),ILABX(5),  MAY11     7
                1         ILABY(5),HEAD3(3)                                       PLT       4
000007           COMMON PJ(50),CJ(50),QJ(50),AJ(50),XJ(50),YJ(50),                MAY11     8
                1A(1500),B(1500),G(3000)                                          MAY11     9
000007           COMMON DD,BR,BA,FJ,CC,YM,NN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M      PLT       6
000007           COMMON QA(50),QB(50),QC(50),QA1(50),QB1(50),QC1(50),QA2(50),QB2(50 PLT     7
                1),QC2(50) ,IH,IP                                                 PLT       8
000007           COMMON WT(50),HEAD1(6),HEAD2(6)                                  MAY1     11
000007           DATA (ILABX(J),J=1,3)/10H    CALCUL,10HATION POIN,2HT /          PLT      10
000007           DATA (ILABY(J),J=1,3)/10HTANGENTIAL,10H ERROR (IN,2H.)/          PLT      11
000007           DATA ILABZ/10HLOAD (LB.)/                                        PLT      12
000007           DATA (HEAD3(I),I=1,2)/10HTANG. FORC,4HE = /                      PLT      13
000007           IN=5                                                             PLT      14
000007           IO=6                                                             PLT      15
000010           ICAL=22                                                          PLT      16
000012           CALL PLOTS(IBUF,4000,ICAL)                                       PLT      17
000014           CALL PLOT(4.,4.5,-3)                                             PLT      18
000017           CALL PLOT(-2.,-4.,3)                                             PLT      19
000022           CALL DASHPT(-2.,6.,.25)                                          PLT      20
000025           CALL SCALE(X,5.,NPTS,+1)                                         PLT      21
000032           Y(NPTS+1)=0.                                                     PLT      22
000036           DO 10 K=1,NPTS                                                   PLT      23
000040        10 Y(K)=-Y(K)                                                       PLT      24
000043           CALL SCALE(Y,4.,NPTS+1,+1)                                       PLT      25
000050           Y(NPTS+2)=Y(NPTS+3)                                              PLT      26
000056           CALL SCALE(Z,3.,NPTS,+1)                                         PLT      27
000061           AXX(1)=X(NPTS+1)                                                 PLT      28
000065           AXX(2)=X(NPTS+2)                                                 PLT      29
000067           AYY(1)=Y(NPTS+1)                                                 PLT      30
000071           AYY(2)=Y(NPTS+2)                                                 PLT      31
000073           AZZ(1)=Z(NPTS+1)                                                 PLT      32
000074           AZZ(2)=Z(NPTS+2)                                                 PLT      33
000077           CALL AXIS(0.,0.,ILABX,-22,5.,0.,AXX(1),AXX(2))                   PLT      34
000106           CALL AXIS(0.,0.,ILABY,22,4.,90.,0.,AYY(2))                       PLT      35
000116           CALL FLINE(X,Y,-NPTS,1,1,3)                                      PLT      36
000125           CALL PLOT(0.,-4.,-3)                                             PLT      37
000130           CALL AXIS(0.,0.,ILABX,-22, 5.,0.,AXX(1),AXX(2))                  PLT      38
000140           CALL AXIS(0.,0.,ILABZ,10,3.,90.,AZZ(1),AZZ(2))                   PLT      39
000150           CALL FLINE(X,Z,-NPTS,1,1,11)                                     PLT      40
000157           CALL PLOT(.5,9.5,-3)                                             PLT      41
000162           CALL SYMBOL(0.,0.,.14,HEAD1,0.,40)                               PLT      42
000166           CALL SYMBOL(0.,-.25,.14,HEAD2,0.,40)                             PLT      43
000172           CALL SYMBOL(0.,-.5,.14,HEAD3,0.,14)                              PLT      44
000176           CALL NUMBER(999.,999.,.14,WT,0.,2)                               PLT      45
000202           CALL SYMBOL(999.,999.,.14,EHEAD3,0.,4)                           PLT      46
000206           CALL PLOT(6.,-9.,-3)                                             PLT      47
000211           CALL DASHPT(0.,9.,.25)                                           PLT      48
000214           CALL PLOT(0.,0.,999)                                             PLT      49
000217           DO 20 K=1,NPTS                                                   PLT      50
000223        20 Y(K)=-Y(K)                                                       PLT      51
000226           RETURN                                                           PLT      52
000227           END                                                              PLT      53
```

NOTE TO USERS OF COMPUTER PROGRAM

The computer program for the prediction of gear mesh excitation spectra has the ability to plot tooth meshing error. This meshing error can be plotted if, and only if, a CALCOMP Plotter is a part of the system's hardware. If this plotting hardware is unavailable, the plotting subroutine (subroutine PLT (X,Y,Z,NPTS)) must be bypassed. This is done by placing a zero in column 25 on card 2 of the input deck.

RUN VERSION 2.3 · ·PSR LEVEL 332--

```
                  SUBROUTINE SPECT                                                  MAY9      14
000002            DIMENSION GN(3000),RR(3000),FR(200),GK(200),KK(200),GKK(200)      MAY11     10
000002            COMMON PJ(50),CJ(50),OJ(50),AJ(50),XJ(50),YJ(50),                 MAY11     11
                 1A(1500),B(1500),G(3000)                                           MAY11     12
000002            COMMON DD,BR,BA,FJ,CC,YM,NN,N,FNJ,EP,TAN,RB,GM,F,XM,MN,II,M       GEARC      1
000002            COMMON/GU/NMC,INT,MMM,IPLT,FN1,FN2,RR1,RO1,RO2,RT1,RT2,RM1,RM2,FI, MAY11    14
                 1T1,T2,F1,F2,RF1,RF2,YE1,YE2,GE1,GE2,POS1,POS2,ANG                 MAY11     15
000002            NR=5                                                              MAY11     16
000003            NW=6                                                              MAY11     17
            C NWS=1,WS=SPEED OF DRIVING GEAR IN RPM                                 MAY11     18
            C NWS=2,WS=SPEED OF DRIVEN GEAR IN RPM                                  MAY11     19
000004            READ(NR,100) NWS,WS                                               MAY11     20
000014            WS=WS*6.28/60.                                                    MAY11     21
000016            FM=FN1*WS/6.28                                                    MAY11     22
000020            IF(NWS.EQ.2) FM=FN2*WS/6.28                                       MAY11     23
000024            WRITE(NW,110) FM                                                  MAY11     24
000032            FC=FM*N/2.                                                        MAY11     25
000035            NT=N*NMC                                                          MAY11     26
000037            TR=NMC/FM                                                         MAY11     27
000041            H=1./FM/N                                                         MAY11     28
000044            FO=1./TR                                                          MAY11     29
000046            M=NT/20                                                           MAY11     30
000051            SM=M                                                              MAY11     31
000052            THAX=M*H                                                          MAY11     32
000055            BE=1./THAX                                                        MAY11     33
000057            WRITE(NW,115)                                                     MAY11     34
000062            WRITE(NW,120) FC,FO,BE                                            MAY11     35
000074            WRITE(NW,125)                                                     MAY11     36
000100            WRITE(NW,120) H,TR,THAX,SM                                        MAY11     37
000114            SUM=0.                                                            MAY11     38
000115            DO 10 I=1,NT                                                      MAY11     39
000117         10 SUM=SUM+G(I)                                                      MAY11     40
000123            SA=SUM/NT                                                         MAY11     41
000125            DO 20 I=1,NT                                                      MAY11     42
000127            GN(I)=G(I)-SA                                                     MAY11     43
000132         20 CONTINUE                                                          MAY11     44
000134            MM=M+1                                                            MAY11     45
000136            DO 30 I=1,MM                                                      MAY11     46
000137            IR=I-1                                                            MAY11     47
000140            FR(I)=IR*FC/SM                                                    MAY11     48
000144            CC=1./(NT-IR)                                                     MAY11     49
000147            RR(I)=0.                                                          MAY11     50
000151            DO 25 J=1,NT                                                      MAY11     51
            C RR(I)=R(I-1)                                                          MAY11     52
000153            IF((J+IR).GT.NT) GO TO 30                                         MAY11     53
000157         25 RR(I)=GN(J)*GN(J+IR)+RR(I)                                        MAY11     54
000165         30 RR(I)=RR(I)*CC                                                    MAY11     55
000172            MS=M-1                                                            MAY11     56
000174            DO 40 K=1,MM                                                      MAY11     57
000175            KK(K)=K-1                                                         MAY11     58
000177            GK(K)=RR(1)+RR(MM)*(-1.)**KK(K)                                   MAY11     59
000206            SSS=0.                                                            MAY11     60
000207            DO 35 JK=1,MS                                                     MAY14      2
000211            JJ=JK+1                                                           MAY14      3
```

```
000212           AA=3.1416*JK*KK(K)/SM                                   MAY11    62
000217        35 SSS=SSS+2.*RR(JJ)*COS(AA)                               MAY11    63
000230        40 GK(K)=(GK(K)+SSS)*2.*H                                  MAY11    64
000235           GKK(1)=.5*GK(1)+.5*GK(2)                                MAY11    65
000241           GKK(MM)=.5*GK(M)+.5*GK(MM)                              MAY11    66
000246           DO 50 LL=2,M                                            MAY11    67
000247           GKK(LL)=.25*GK(LL-1)+.5*GK(LL)+.25*GK(LL+1)             MAY11    68
000255        50 CONTINUE                                                MAY11    69
000260           WRITE(NW,130)                                           MAY11    70
000264           WRITE(NW,140)                                           MAY11    71
000270           DO 60 I=1,MM                                            MAY11    72
000272           SKM=KK(I)*N/(2.*SM)                                     JUNE7     6
000277           WRITE(NW,150)KK(I),FR(I),GK(I),GKK(I),SKM               JUNE7     7
000314        60 CONTINUE                                                MAY11    74
000317       100 FORMAT(I5,E13.5)                                        MAY11    75
000317       110 FORMAT(/1X26HMESHING FREQUENCY IN CPS =,E13.5)          MAY11    76
000317       115 FORMAT(/7X2HFC11X2HFO11X2HBE)                           MAY11    77
000317       120 FORMAT(8E13.5)                                          MAY11    78
000317       125 FORMAT(/7X1HM12X2HTR11X4HTMAX9X2HSM)                    MAY11    79
000317       130 FORMAT(/1X31HPOWER SPECTRAL DENSITY FUNCTION)           MAY11    80
000317       140 FORMAT(/6X,1HK6X2HFR11X2HGK11X3HGKK10X2HKM)             JUNE7     8
000317       150 FORMAT(I7,3E13.5,F13.5)                                 JUNE7     9
000317           RETURN                                                  MAY11    83
000317           END                                                     MAY11    84
```

APPENDIX D

LISTING OF DYNAMIC GEAR TOOTH FORCE ANALYSES - TORRP (R-32)

```
C     MEMBERNAME R32AMAIN
C
C     PROGRAM TORRP
C     MECHANICAL TECHNOLOGY INCORPORATED, LATHAM, NEW YORK
C     AVLABS COPY NO.1,1-29-70, RHB
C     THIS PROGRAM REQUIRES SUBROUTINES PLANST,MATIN,BLOOP,CDIV,CDIV2,
C         PANGF,AMPF,CADD,CSUB,CMPY
      DIMENSION AXY(20),BXY(20) ,LEX(20)
      DIMENSION BRAT(2),TEMPS(2),LMC(20),LTW(20),LBST(20)
      COMMON TBR(2,2),THE(2,2),TPR(2,2),TPF(2,2),TFSPR(2,2),TFSPE(2,2),
     1 TFPRR(2,2),TFPRE(2,2),TTPCR(2,2),TTPCE(2,2),TTGRR(2,2),TTGRE(2,2)
      COMMON RIP(200),DST(200),DMS(200),DIN(200),RCB(50),LB(50),DDT(200)
     1,RK(50),LEC(2),RS(2),RW(2),PIP(2),PMS(2),PN(2),SS(2),SW(2),SR(2),
     2 ST(2),LBR(20),LBS(20),CCOM(200),AXY1,AXY2,BXY1,BXY2,TLR(200),
     3 TLE(200),THR(200),THE(200),TRR(200),TRE(200),TRR1(200)
      COMMON A(7,7),B(7,2),LS(20),RP(20),RG(20),SG(20),TRR2(200),
     1 TRE1(200),THE1(200),TLE1(200),TRE2(200),TTFR(20),TTFE(20),
     2 PSP(2),THR1(200),TLR1(200),THR2(200),TLR2(200),THE2(200),
     3 TLE2(200),RL(200),PCP(2),PRP(2)   ,DMP(50)
      COMMON IT, K4,L4,MDIAG,FRQ2,ANGR,ANGE,TQR,TQE ,J,NS ,M1,NW,NR,NPLG
      COMMON  K1,K2,K3,K9,L1,L2,L3,L9,LG(10),IBRAN
C   NR REFERS TO INPUT UNIT
C   NW REFERS TO OUTPUT UNIT
      NR=5
      NW=6
      MDTW=0
      MTH=0
      MDSG=0
      IEXP=0
  200 READ(NR,100)
      READ(NR,101) NS,NB,NBR,NMPG,INP,NSRG,NPLG,MDIAG
      IOP=0
      FFQ=0.0
      CXY=0.
      DXY=0.
      MPOL = -1
      IF (MDIAG -2) 3021,3020,3021
 3020 MPOL = 0
      GO TO 3025
 3021 IF(MDIAG +2) 3025,3022,3025
 3022 MPOL = 1
 3025 READ (NR,102)GM,DENST
      WRITE(NW,100)
      WRITE(NW,103)
      WRITE(NW,106)NS,NB,NBR,NMPG,INP,NSRG,NPLG
 2022 WRITE(NW,135) GM,DENST
      DENST=DENST/386.069
      IF(IEXP) 2023,2024,2025
 2023 IEXP=-IEXP
      AINT=1./10.**IEXP
      GO TO 2026
 2024 AINT=1.
      GO TO 2026
 2025 AINT=10.**IEXP
 2026 WRITE(NW,107)
      DO 201 J=1,NS
      READ(NR,190)NSTA,RIP(J),RL(J),DST(J),DMS(J),DIN(J),CCOM(J)
      IF(NSTA-J) 2000,2001,2000
 2000 WRITE(NW,191)
 2001 WRITE(NW,108)NSTA,RIP(J),RL(J),DST(J),DMS(J),DIN(J),CCOM(J)
```

```
201 RIP(J)=RIP(J)/386.069
    LFX(1) = NS+2
    LB(1)=NS+2
    LFC(1)=NS+2
    LOUT =0
    LF=0
    KL=0
    LI=0
    LBR(1)=NS+2
    LBS(1)=NS+2
    LS(1) =NS+2
    LG(1)=NS+2
    LFT=0
    KLT=0
    LIT=0
    LTW(1)=NS+2
    IF(NB) 2031,2031,203
203 WRITE(NW,109)
    DO 204 J=1,NB
    READ(NR,190)LB(J),BK(J),BCB(J),DMP(J)
204 WRITE(NW,108)LB(J),BK(J),BCB(J),DMP(J)
2031 IF(NSRG)206,206,2032
2032 WRITE(NW,112)
    WRITE(NW,113)
    DO 205 I=1,NSRG
    READ(NR,190)LS(I),RP(I),RG(I),SG(I)
205 WRITE(NW,108)LS(I),RP(I),RG(I),SG(I)
206 IF(NPLG)210,210,209
209 WRITE(NW,114)
    DO 212 J=1,NPLG
    READ (NR,190) LEC(J),PN(J),RS(J),RW(J)
    WRITE(NW,127)
    WRITE (NW,108) LEC(J),PN(J),RS(J),RW(J)
    READ (NR,190) IPL,PMS(J),PSP(J),PIP(J),PRP(J),PCP(J)
    WRITE (NW,128)
    WRITE (NW,108) IPL,PMS(J),PSP(J),PIP(J),PRP(J),PCP(J)
    READ (NR,130) IPL,SS(J),SR(J),SW(J),ST(J)
    WRITE (NW,129)
    WRITE (NW,131) IPL,SS(J),SR(J),SW(J),ST(J)
    PSP(J) =PSP(J)/386.069
    PCP(J)=PCP(J)/386.069
    PRP(J)=PRP(J)/386.069
    PIP(J)=PIP(J)/386.069
212 PMS(J)=PMS(J)/386.069
210 IF(NBR) 2130,2130,214
214 WRITE(NW,118)
    DO 215 J=1,NBR
    READ(NR,101) LBR(J),LBS(J),LMC(J)
215 WRITE(NW,119) LBR(J),LBS(J)
2130 K5=NS-1
    DO 216 J=1,K5
    C1=RL(J)
    C2=DST(J)
    IF(C1) 217,216,217
217 IF(C2) 218,216,218
218 C5=DIN(J)*DIN(J)
    C4=C5*C5
    C6=DMS(J)*DMS(J)
    C8=0.09817477*(C2**4-C4)
    C3=C8*GM
    C4=0.09817477*DENST*(C6*C6-C4)
    DST(J)=C1/C3
    DMS(J) = C1*SQRT(C4/C3)
    DIN(J) = SQRT(C3*C4)
216 CONTINUE
    DO 501 IERR =1,NMPG
    FF0=0.0
    AXY1=0.0
    BXY1=0.0
```

```
      AXY2=0.0
      BXY2=0.0
      MAX =1
      IEX = 1
 4009 READ (NR,190) IT,TE1,TE2,TE3,TE4,TE5,TE6,NEX
      FFQ=TE1
      CXY=TE2
      DXY=TE3
      IF (NEX) 2013,4010,4010
 4010 IF(IEX-1) 4012,4011,4012
 4011 MAX = NEX
 4012 DO 2011 I=1,NSRG
      IF(IT-LS(I)) 2011,2010,2011
 2010 AXY(IEX) = TE2
      BXY(IEX) = TE3
      LEX(IEX) = IT
      WRITE (NW,194)
      WRITE(NW,108)IT,TE1,TE2,TE3
      IEX = IEX +1
      IF (IEX- MAX) 4009,4009,2020
 2011 CONTINUE
      IF(NPLG) 3019,3019,2013
 2013 DO 2017 I=1,NPLG
      IF(IT-LEC(I)) 2017,2014,2017
 2014 WRITE(NW,195)
      AXY1=TE2
      BXY1=TE3
      AXY2=TE4
      BXY2=TE5
      WRITE(NW,108)IT,TE1,TE2,TE3,TE4,TE5
      GO TO 2020
 2017 CONTINUE
 3019 WRITE(NW,192)
      IF(INP) 506,200,506
 2020 FRQ=6.2831853*FFQ
      IF(FRQ) 501,501,599
  599 FRQ2=FRQ*FRQ
      NEX=1
      ISEP=0
 5020 BRAT(1)=1.0
      BRAT(2)=0.0
      IBRAT=0
      IBRAN=0
      M1=1
  502 ANGR=0.0
      IF(IOP) 5021,5022,5021
 5021 ANGR=AINT
 5022 ANGE=0.0
      ANGE=0.0
      TOR=0.0
      TOE=0.0
      A13R=0.0
      A13E=0.0
      K1=1
      K2=1
      K4=1
      K5=NS+2
      K9=1
      L1=LBR(1)
      L2=LB(1)
      K3=1
      L3=LS(1)
      L4 =LEC(1)
      L5=LS(1)
      GO TO (227,228),M1
  227 ANGR=AINT
  228 J=1
      IF(NTW) 2275,2275,2270
 2270 J=LTW(1)
```

```
      LIT=1
      KLT=1
 2275 IF(NBR) 2280,2280,2279
 2279 LF=LBR(1)
      LOUT=LMC(1)
      IF(LOUT) 2277,2278,2278
 2277 LOUT=-LOUT
 2278 LI=1
      KL=1
 2280 IF(IOP) 2282,2281,2282
 2281 IF(J-LF) 249,2300,249
 2300 CALL  BLOOP(LBR,LBS,LMC,LI,KL,LF,NBR,LIT,KLT,LFT)
      GO TO 249
 2282 IF(J-L1) 231,230,231
  230 IF(M1-1) 232,232,234
  234 IBRAN=1
  232 TL1R=TOR
      AL1R=ANGR
      TL1E=TOE
      AL1E=ANGE
  562 ANGR=AINT
      ANGE=0.
      TOR=0.
      TOE=0.
      A13R=0.
      A13E=0.
  235 K5=LBS(K1)
      K6=L1
      K1=K1+1
      IF(NBR-K1) 237,236,236
  236 L1=LBR(K1)
      GO TO 249
  237 L1=NS+2
      GO TO 249
  231 IF(J-K5) 249,248,249
  248 AA1 =(ANGR*ANGR+ANGE*ANGE)
      IBRAN=0
      AA2= SQRT(AA1)
      AA5 =AL1R*ANGR +AL1E*ANGE
      AA3 =(AL1R*AL1R +AL1E*AL1E)
      AA4= SQRT(AA3)
      IF (AA2-1.0E-7*AA4) 242,241,241
  241 AAR =AA5/AA1
      AAE = (ANGR*AL1E- ANGE*AL1R) /AA1
      TE1 = TOR
      TE2 = TL1R
      TE3 = TOE
      TE4 = TL1E
      ANGR= AL1R
      ANGE = AL1E
      K7=K6
      K8=K5-1
      IF(M1-2) 2420,2410,2420
 2410 IF(IBRAT) 2411,2420,2411
 2411 IF(IBRAT-2) 2420,2412,2420
 2412 BRAT(1)=AAR
      BRAT(2)=AAE
 2413 IBRAT=1
      GO TO 2420
  242 AAR = AA5/ AA3
      AAE = ( AL1R*ANGE- AL1E*ANGR) / AA3
      TE1 = TL1R
      TE2 = TOR
      TE3 =TL1E
      TE4 = TOE
      K7=1
      K8=K6-1
      IF (M1-2) 2420,2440,2420
 2440 IF(IBRAT) 2441,2420,2441
 2441 IF(IBRAT-1) 2413,2412,2413
```

```
 2420 TQR = AAR* TE1 -AAE* TE3 +TE2
      TQE = AAE *TE1 + AAR*TE3 + TE4
  243 DO 244 L=K7,K8
      C1 = TLR(L)
      C2 = TLE(L)
      C3 = THR(L)
      C4 = THE(L)
      C5 = TRR(L)
      C6 = TRE(L)
      TLR(L) = AAR*C1-AAE*C2
      THR(L) = AAR*C3-AAE*C4
      TRR(L) = AAR*C5-AAE*C6
      TLE(L) = AAE*C1+ AAR* C2
      THE(L) = AAE*C3+ AAR* C4
  244 TRE(L) = AAE*C5+ AAR* C6
  249 THE(J)=ANGE
      TLE(J)=TQE
      THR(J)=ANGR
      TLR(J)=TQR
      CD=0.0
      IF(J-L2) 284,283,284
  283 C1=RK(K2)
      CD=DMP(K2)
      C2=RCB(K2)
      K2=K2+1
      IF(NR-K2) 261,260,260
  260 L2=LR(K2)
      GO TO 285
  261 L2=NS+2
      GO TO 285
  284 C1=0.0
      C2=0.0
  285 C1=C1-FRQ2*RIP(J)
      C2=FRQ*C2
      TQE=TQE+C1*ANGE   +C2*ANGR
      TQR=TQR+C1*ANGR-C2*ANGE
      IF(J-LOUT) 2853,2850,2853
 2850 J1=LBR(KL-1)
      TQR=TQR-TLR(J1)
      TQE=TQE-TLE(J1)
      IF(KL-NBR) 2851,2851,2852
 2851 LOUT=LMC(KL)
      IF(LOUY) 2848,2853,2853
 2848 LOUT=-LOUT
      GO TO 2853
 2852 LOUT=NS+2
 2853 TRE(J)=TQE
      TRR(J)=TQR
      IF(J-NS) 286,287,287
  287 IF(M1-1) 401,401,402
  286 IF(J-L3) 288,289,288
  289 C1=RP(K3)
      C2=RG(K3)
      A11=-C1/C2
      A22=1.0/A11
      A21=0.0
      A12=-SG(K3)/(C1*C2)
      K3=K3+1
      IF(K3-NSRG) 1019,1019,1020
 1019 L3=LS(K3)
      GO TO 1021
 1020 L3=NS+2
 1021 IF (M1-1) 351,351,251
  251 IF (LEX(NEX) - J) 351,2899,351
 2899 IF(M1-2) 290,2900,2900
 2900 IF(IBRAN) 2901,2910,2901
 2901 IF(IBRAT) 2902,2903,2902
 2902 WRITE(NW,136)
      ISEP=1
      GO TO 351
```

```
 2903 IBRAT=2
      GO TO 290
 2910 IF(IBRAT) 2912,2911,2912
 2911 IBRAT=1
      GO TO 290
 2912 IF(BRAT(2)-1.0) 2902,2913,2902
 2913 IF(BRAT(2)) 2902,290,2902
  290 A13R=-AXY(NEX)/C2
      A13E=-BXY(NEX)/C2
      IF (NEX-MAX) 5000,351,351
 5000 NEX = NEX+1
      GO TO 351
  288 IF(J-L4) 255,254,255
  254 IF(MDIAG+1) 1092,1092,1095
 1092 IF (K4-1) 1095,1096,1095
 1096 IF (M1-1) 1093,1093,1094
 1093 WRITE (NW,181)
      GO TO 1095
 1094 WRITE (NW,182)
 1095 CALL PLNST
      GO TO 503
  255 C1=RL(J)
      IF(C1) 257,256,257
  256 A11=1.0
      A12 =CCOM(J)
      IF(CD) 3001,3000,3001
 3000 A21=0.0
      A22=1.0
      GO TO 351
 3001 TEMP=1.0+(FRQ*CD*A12)**2
      TEMP1=FRQ*CD*A12**2
      ANGR=ANGR+(A12*TQR+TEMP1*TQE)/TEMP
      ANGE=ANGE+(A12*TQE-TEMP1*TQR)/TEMP
      GO TO 503
  257 C3=FRQ*DMS(J)
      IF(C3-0.0003) 258,258,259
  258 C4=C3*C3
      A11=1.0-0.5*C4
      A22=A11
      A12=DDT(J)
      A21=-C3*DIN(J) *FRQ
      GO TO 351
  259 A11= COS(C3)
      A22=A11
      C5=DIN(J)*FRQ
      C4= SIN(C3)
      A12=C4/C5
      A21=-C4*C5
  351 C1=ANGE
      ANGE=A11*ANGE+A12*TQE+A13 E
      TQE=A21*C1   +A22*TQE
      C1=ANGR
      ANGR=A11*ANGR+A12*TQR+A13R
      TQR=A21*C1   +A22*TQR
      A13R=0.0
      A13E=0.0
  503 J=J+1
      IF(J-NS) 2280,2280,401
  401 IF(MDIAG+1) 410,410,411
  410 IF (NPLG) 1090,1090,1091
 1090 WRITE(NW,18))
 1091 WRITE(NW,126)
      WRITE(NW,701)
  411 DO 403 J=1,NS
      TRR1(J)=TRR(J)
      THR1(J)=THR(J)
      TLR1(J)=TLR(J)
      TRE1(J)=TRE(J)
      THE1(J)=THE(J)
      TLE1(J)=TLE(J)
```

```
      IF(MDIAG+1) 1030,1030,403
 1030 WRITE(NW,108)J,THR1(J),THE1(J),TLR1(J),TLE1(J),TRR1(J),TRE1(J)
  403 CONTINUE
      M1=M1+1
      GO TO 50
  402 IRAT=1
      IF(BRAT(1)-1.0) 4132,4130,4132
 4130 IF(BRAT(2)) 4132,4133,4132
 4132 IRAT=2
 4133 IF(MDIAG+1) 412,412,413
  412 IF(NPLG) 414,414,415
  414 WRITE(NW,182)
  415 WRITE(NW,126)
      WRITE(NW,701)
  413 DO 404 J=1,NS
      IF(IRAT-2) 4040,4041,4040
 4040 TLR2(J)=TLR(J)
      THE2(J)=THE(J)
      TLE2(J)=TLE(J)
      TRR2(J)=TRR(J)
      TRE2(J)=TRE(J)
      THR2(J)=THR(J)
      GO TO 4045
 4041 TEMPS(1)=TLR(J)
      TEMPS(2)=TLE(J)
      CALL CDIV(TEMPS,BRAT,TEMPS)
      TLR2(J)=TEMPS(1)
      TLE2(J)=TEMPS(2)
      TEMPS(1)=THR(J)
      TEMPS(2)=THE(J)
      CALL CDIV(TEMPS,BRAT,TEMPS)
      THR2(J)=TEMPS(1)
      THE2(J)=TEMPS(2)
      TEMPS(1)=TRR(J)
      TEMPS(2)=TRE(J)
      CALL CDIV(TEMPS,BRAT,TEMPS)
      TRR2(J)=TEMPS(1)
      TRE2(J) =TEMPS(2)
 4045 IF(MDIAG+1) 1040,1040,404
 1040 WRITE(NW,108)J,THR2(J),THE2(J),TLR2(J),TLE2(J),TRR2(J),TRE2(J)
  404 CONTINUE
      IF(IRAT-2) 4050,4046,4050
 4046 IF(NPLG) 4050,4050,4047
 4047 DO 4048 KK=1,NPLG
      CALL CDIV2(TBR,TBE,BRAT,KK,MDIAG)
      CALL CDIV2(TPR,TPE,BRAT,KK,MDIAG)
      CALL CDIV2(TFSPR,TFSPE,BRAT,KK,MDIAG)
      CALL CDIV2(TFPRR,TFPRE,BRAT,KK,MDIAG)
      CALL CDIV2(TTPCR,TTPCE,BRAT,KK,MDIAG)
 4048 CALL CDIV2(TTGRR,TTGRE,BRAT,KK,MDIAG)
 4050 C1=TRR2(NS)
      C2=TRE2(NS)
      C3=TRR1(NS)
      C4=TRE1(NS)
      C5=C4/C3
      C6=C3+C4*C5
      Q1R=-(C1+C5*C2)/C6
      Q1E=(C1*C5-C2)/C6
      IF (MDIAG-2) 1049,1051,1049
 1049 IF(MDIAG) 1050,1051,1050
 1050 WRITE(NW,183)
      WRITE(NW,126)
      WRITE(NW,701)
 1051 L=1
      DO 407 J=1,NS
      C1=THR1(J)*Q1R+THR2(J) -Q1E*THE1(J)
      C2=THR1(J)*Q1E+THE2(J)+Q1R*THE1(J)
      C3=TLR1(J)*Q1R+TLR2(J) -Q1E*TLE1(J)
      C4=TLR1(J)*Q1E+TLE2(J)+TLE1(J)*Q1R
      C5     =TRR1(J)*Q1R-Q1E*TRE1(J)+TRR2(J)
```

```
      C6    =TRR1(J)*Q1E+Q1R*TRE1(J)+TRE2(J)
      NGS =LS(L)
      IF(J-NGS) 1054,1100,1054
 1100 TTFR(L) =C5/RP(L)
      TTFE(L) =C6/RP(L)
      L =L+1
 1054 IF(MDIAG) 1055,407,1055
 1055 IF(MPOL) 1056,1057,1056
 1056 WRITE(NW,108)J,C1,C2,C3,C4,C5,C6
      IF(MPOL) 407,1057,1057
 1057 THR1(J)=AMPF(C1,C2)
      THE1(J)=PANGF(C1,C2)
      TLR1(J)=AMPF(C3,C4)
      TLE1(J)=PANGF(C3,C4)
      TLR2(J)=AMPF(C5,C6)
      TLE2(J)=PANGF(C5,C6)
  407 CONTINUE
C TEST FOR POLAR FORM OPTION
      IF(MPOL) 1195,1058,1058
 1058 WRITE(NW,126)
      WRITE(NW,123)
      DO 1059 J=1,NS
 1059 WRITE(NW,108)J,THR1(J),THE1(J),TLR1(J),TLE1(J),TLR2(J),TLE2(J)
 1195 IF(NPLG) 1201,1201,1196
 1196 IF(MDIAG) 1199,1201,1199
 1199 DO 1200 J=1,NPLG
      C7 =PN(J) *RS(J)
      C8 =( RS(J)  +2.  *RW(J))*PN(J)
      C1 =TBR(1,J)*Q1R+TBR(2,J)-Q1E*TBE(1,J)
      C2 =TBR(1,J)*Q1E+TBE(2,J)+Q1R*TBE(1,J)
      C3 =C7*(TFSPR(1,J)*Q1R+TFSPR(2,J)-Q1E*TFSPE(1,J))
      C4 =C7*(TFSPR(1,J)*Q1E+TFSPE(2,J)+Q1R*TFSPE(1,J))
      C5 =TTPCR(1,J)*Q1R+TTPCR(2,J)-Q1E*TTPCE(1,J)
      C6 =TTPCR(1,J)*Q1E+TTPCE(2,J)+Q1R*TTPCE(1,J)
      IF(MPOL) 1060,1061,1060
 1060 WRITE(NW,700)
      WRITE(NW,701)LEC(J),C1,C2,C3,C4,C5,C6
      IF(MPOL) 1063,1061,1061
 1061 WRITE(NW,700)
      JP=1
 1062 TE1=AMPF(C1,C2)
      TE2=PANGF(C1,C2)
      TE3=AMPF(C3,C4)
      TE4=PANGF(C3,C4)
      TE5=AMPF(C5,C6)
      TE6=PANGF(C5,C6)
      WRITE(NW,123)
      WRITE(NW,108)LEC(J),TE1,TE2,TE3,TE4,TE5,TE6
      GO TO (1063,1200),JP
 1063 C1=TPR(1,J)*Q1R+TPR(2,J)-Q1E*TPE(1,J)
      C2 =TPR(1,J)*Q1E+TPE(2,J)+Q1R*TPE(1,J)
      C3 =C8*(TFPRR(1,J)*Q1R+TFPRR(2,J)-Q1E*TFPRE(1,J))
      C4 =C8*(TFPRR(1,J)*Q1E+TFPRE(2,J)+Q1R*TFPRE(1,J))
      C5 =TTGRR(1,J)*Q1R+TTGRR(2,J)-Q1E*TTGRE(1,J)
      C6 =TTGRR(1,J)*Q1E+TTGRE(2,J)+Q1R*TTGRE(1,J)
      IF(MPOL) 1064,1065,1064
 1064 WRITE(NW,702)
      WRITE(NW,701)LEC(J),C1,C2,C3,C4,C5,C6
      IF(MPOL) 1200,1065,1065
 1065 JP=2
      WRITE(NW,702)
      GO TO 1062
 1200 CONTINUE
 1201 IF(NSRG) 1207,1207,1202
 1202 IF(MPOL) 1203,1205,1203
 1203 WRITE(NW,180)
      DO 1070 J=1,NSRG
 1070 WRITE (NW,108) LS(J),TTFR(J),TTFE(J)
      IF(MPOL) 1207,1205,1205
 1205 WRITE(NW,124)
```

```
      DO 1206 J=1,NSRG
      C1=AMPF(TTFR(J),TTFE(J))
      C2=PANGF(TTFR(J),TTFE(J))
 1206 WRITE(NW,108)LS(J),C1,C2
 1207 IF(NPLG) 1219,1219,1079
 1079 IF(MPOL) 1210,1215,1210
 1210 WRITE(NW,122)
 1215 DO 1080 J=1,NPLG
      TE1=TFSPR(1,J)
      TE2=TFSPR(2,J)
      TE3=TFSPE(1,J)
      TE4=TFSPE(2,J)
      C1=TE1*Q1R+TE2-TE3*Q1E
      C2=TE1*Q1E+TE4+TE3*Q1R
 1076 TE1=TFPRR(1,J)
      TE2=TFPRR(2,J)
      TE3=TFPRE(1,J)
      TE4=TFPRE(2,J)
      C3=TE1*Q1R+TE2-TE3*Q1E
      C4=TE1*Q1E+TE4+TE3*Q1R
      IF(MPOL) 1216,1077,1216
 1077 THR1(J)=AMPF(C1,C2)
      THE1(J)=PANGF(C1,C2)
      TLR1(J)=AMPF(C3,C4)
      TLE1(J)=PANGF(C3,C4)
      GO TO 1080
 1216 WRITE(NW,108)LEC(J),C1,C2,C3,C4
 1080 CONTINUE
      IF(MPOL) 1219,1217,1217
 1217 WRITE(NW,125)
      DO 1218 J=1,NPLG
 1218 WRITE(NW,108) LEC(J),THR1(J),THE1(J),TLR1(J),TLE1(J)
 1219 IF(NEX-MAX) 5007,501,501
 5007 IF(ISEP-1) 5009,5008,5009
 5008 ISEP=2
      GO TO 5020
 5009 NEX=NEX+1
      IF(NEX-MAX) 5020,501,501
  501 CONTINUE
 5010 IF(INP) 506,200,506
  506 CALL EXIT
  100 FORMAT(72H1
     1                                                                        )
  101 FORMAT(9I5,5X,3I5,10X,I5)
  102 FORMAT(5X,6E12.4)
  103 FORMAT(82H0 TORSIONAL RESPONSE OF THE SYSTEM WITH GEARS,EPICYCLIC
     1GEARS AND BRANCHES  PN408 )
  105 FORMAT(8X,6E12.5)
  106 FORMAT(1H0,1X,8HSTATIONS,1X,12HBRG+EXT.CON.,2X,8HBRANCHES,1X,
     19HNO.OF FRQ,2X,8HINP SETS,2X,8HSR GEARS,2X,8HPL GEARS/
     2         I7,6X,I5,2X,5I10)
  107 FORMAT(/5X,10HROTOR DATA//4X,3HSTA,1X,12H  MOM.INERT.,3X,
     116HLENGTH,4X,10HSTIFFN.DIA,3X,8HMASS DIA,3X,9HINNER DIA,2X,
     212HCONC. COMPL./4X3HNO,2X,9HLBS-IN**2,7X,2HIN,10X,2HIN,10X,2HIN,
     310X,2HIN,6X, 9HRAD/IN-LB)
  108 FORMAT(I7,1X,6E12.5)
  109 FORMAT(/1H1,7X,46HEXTERNAL TORSIONAL CONSTRAINT AND DAMPING DATA//
     1 4X,3HSTA,2X, 9HSTIFFNESS,2X,24HDAMPING- (LB-IN-SEC/RAD)/
     2 4X,3HNO.,1X,11H(LB-IN/RAD),2X,9HTO GROUND,2X,12HTO NEXT STA.)
  112 FORMAT(/1H1,9X,26HSINGLE REDUCTION GEAR DATA)
  113 FORMAT(/4X,3HSTA,3X,20HGEAR SET RADII (IN.),2X,
     1 31HCOMBINED TANGENTIAL COMPLIANCES//9X,10HFIRST GEAR,2X
     2 ,11HSECOND GEAR,3X,7H(IN/LB))
  114 FORMAT (/15X,18HPLANETARY SET DATA)
  118 FORMAT(/5X,17HBRANCHES         /6X,7HFAR END,5X,6HCOMMON)
  119 FORMAT(6X,3(I5,5X))
  120 FORMAT ( 1H0,5X,11HFREQUENCY =,E14.6,2X,3HCPS)
  121 FORMAT(/1H1)
  122 FORMAT(/ 8X,40HCOMPUTED RESPONSE AT PLANETARY GEAR SETS//
     1 10X,43HTANGENTIAL TOOTH FORCE AT EACH PLANET (LBS)/
     2 4X,3HSTA,8X,10HSUN-PLANET,13X,11HRING-PLANET/
```

```
     3 4X,3HNO.,5X,4HREAL,6X,9HIMAGINARY,5X,4HREAL,6X,9HIMAGINARY)
 123 FORMAT(9X,9HAMPLITUDE,1X,13HPHASE-ANG,DEG,1X,9HAMPLITUDE,1X,
     1 13HPHASE-ANG,DEG,1X,9HAMPLITUDE,1X,13HPHASE-ANG,DEG)
 124 FORMAT(/8X,36HCOMPUTED RESPONSE AT SIMPLE GEAR SET/
     1/ 4X,3HSTA,2X,28HTANGENTIAL TOOTH FORCE (LBS)/  9X,9HAMPLITUDE,
     2 1X ,13HPHASE-ANG,DEG)
 125 FORMAT(/ 8X,40HCOMPUTED RESPONSE AT PLANETARY GEAR SETS//
     1 10X,43HTANGENTIAL TOOTH FORCE AT EACH PLANET (LBS)/
     2 4X, 3HSTA,8X,10HSUN-PLANET,13X,11HRING-PLANET/ 4X,3HNO.,2X,
     3 9HAMPLITUDE,1X,13HPHASE-ANG,DEG,1X,9HAMPLITUDE,1X,
     4 13HPHASE-ANG,DEG)
 126 FORMAT(/ 4X,3HSTA,4X,19HANGULAR DISP.(RAD.)  ,5X, 8HTORQUE-1,2X,
     1 8H(IN-LBS),6X, 8HTORQUE-2,2X,8H(IN-LBS))
 127 FORMAT(/4X,3HSTA,4X,6HNO. OF,8X,3HSUN,7X,6HPLANET/
     14X,3HNO.,3X,7HPLANETS,4X,10HRADIUS(IN),2X,10HRADIUS(IN))
 128 FORMAT(/4X,3HSTA,2X,10HWEIGHT(LB),4X,41HPOLAR MASS MOMENTS OF INER
     1TIA (LBS-IN**2)/ 4X,3HNO.,4X,6HPLANET,8X,3HSUN,7X,6HPLANET,7X,
     24HRING,7X,7HCARRIER)
 129 FORMAT(/4X,3HSTA,6X,26HCOMPLIANCE- LINEAR(IN./LB),16X,
     1 15HCOM-ANG(RAD/LB)/ 4X,3HNO.,2X,10HSUN-PLANET,2X,11HPLANET-RING,
     2 1X,10HPLAN.-CAR.,14X,11HRING-GROUND)
 130 FORMAT (I5,3F12.4,12X,F12.4)
 131 FORMAT(I7,1X,3E12.4,12X,F12.4)
 132 FORMAT(/5X,13HBRANCH OPTION,I5)
 135 FORMAT(/9X,10HSHEAR MOD.,2X,11HWT. DENSITY/9X, 9HLBS/IN**2 ,3X,
     1 9HLBS/IN**3/8X,2E12.5)
 136 FORMAT(/5X,35HADDED EXCITATION TREATED SEPARATELY)
 180 FORMAT (/8X,36HCOMPUTED RESPONSE AT SIMPLE GEAR SET/
     1/4X,3HSTA,2X,28HTANGENTIAL TOOTH FORCE (LBS)/12X,4HREAL,6X,
     2 9HIMAGINARY)
 181 FORMAT(12H0DIAGNOSTICS/5X,53HFIRST PASS- UNIT AMPLITUDE AT STATION
     1 1-NO EXCITATION)
 182 FORMAT(12H0DIAGNOSTICS/5X,57HSECOND PASS- ZERO AMPLITUDE AT STATIO
     1N 1- WITH EXCITATION)
 183 FORMAT(42H0OUTPUT- COMPUTED RESPONSE AT ALL STATIONS/
     110X,63H(TORQUE-1- GOING INTO STATION. TORQUE-2- COMING OUT OF STAT
     2ION))
 184 FORMAT(8X,6HTFSPR1,6X,6HTFSPR2,6X,6HTFSPE1,6X,6HTFSPE2,9X,3HQ1R,
     19X,3HQ1E)
 185 FORMAT(8X,6HTFPRR1,6X,6HTFPRR2,6X,6HTFPRE1,6X,6HTFPRE2)
 190 FORMAT (I5,6E12.4,I3)
 191 FORMAT(38H0STATION NO. INCORRECT OR OUT OF ORDER)
 192 FORMAT( 37H0STATION NO. FOR GEAR ERROR INCORRECT)
 194 FORMAT(1H1,14X,46HSINGLE REDUCTION GEAR- LINEAR EXCITATION (IN.)/
     14X,3HSTA,2X,9HFREQUENCY,6X,4HREAL,6X,9HIMAGINARY)
 195 FORMAT(1H1,14X,39HPLANETARY GEAR- LINEAR EXCITATION (IN.)/
     14X,3HSTA,2X,9HFREQUENCY,9X, 8HSUN GEAR,17X, 9HRING GEAR/
     224X,4HREAL,6X,9HIMAGINARY,5X,4HREAL,6X,9HIMAGINARY)
 700 FORMAT(/4X,3HSTA,1X,24HANG. DISP.-PLANET CENTER,1X,
     1 22HTOT. TORQUE SUN-PLANET,2X,23HTOT TORQUE PLAN-CARRIER)
 701 FORMAT( 4X,3HNO.,5X,4HREAL,6X,9HIMAGINARY,5X,4HREAL,6X,9HIMAGINARY
     15X,4HREAL,6X,9HIMAGINARY /1/,1X,6E12.5)
 702 FORMAT(/ 4X,3HSTA,2X,22HANG. DISP.-PLANET BODY,2X,
     1 23HTOT. TORQUE PLANET-RING,2X,22HTOT. TORQUE GRND.-RING)
     END
     FUNCTION AMPF(A,B)
C      MEMBERNAME R32AMPF
     AMPF=SQRT(A**2+B**2)
     RETURN
     END
     SUBROUTINE BLOOP(LFE,LOC,LMC,LI,KL,LF,NBR,LIT,KLT,LFT)
C
C      MEMBERNAME R32BLOOP
C
     DIMENSION LT1(2),LT2(2),LT3(2),LT4(2),LT9(2),KT1(2),KT2(2),KT3(2)
     1, KT4(2),KT9(2),MLIT(2),MKLT(2),MLFT(2)
     DIMENSION LFE(20),LOC(20),LMC(20),TC(2),OC(2),TCT(2),OCT(2),TCZ(2)
     1,OCZ(2),TMZ(2),OMZ(2),TM1(2),TBCZ(2),OM(2),TBM(2),TF(2),OM1(2),
     2TCTH(2),OCTH(2),TEMP1(2),TEMP2(2),OF(2)
     COMMON TBR(2,2),TBE(2,2),TPR(2,2),TPE(2,2),TFSPR(2,2),TFSPE(2,2),
     1 TFPRR(2,2),TFPRE(2,2),TTPCR(2,2),TTPCE(2,2),TTGRR(2,2),TTGRE(2,2)
```

```
      COMMON RIP(200),DST(200),DMS(200),DIN(200),BCR(50),LB(50),DDT(200)
     1,BK(50),LFC(2),RS(2),RW(2),PIP(2),PMS(2),PN(2),SS(2),SW(2),SR(2),
     2 ST(2),LBR(20),LBS(20),CCOM(200),AXY1,AXY2,BXY1,BXY2,TLR(200),
     3 TLE(200),THR(200),THE(200),TRR(200),TRE(200),TRR1(200)
      COMMON A(7,7),B(7,2),LS(20),RP(20),RG(20),SG(20),TRR2(200),
     1 TRE1(200),THE1(200),TLE1(200),TRE2(200),TTFR(20),TTFE(20),
     2 PSP(2),THR1(200),TLR1(200),THR2(200),TLR2(200),THE2(200),
     3 TLE2(200),RL(200),PCP(2),PRP(2)   ,DMP(50)
      COMMON IT, K4,L4,NDIAG,FRQ2,ANGR,ANGE,TQR,TQE  ,J,NS  ,M1,NW,NR,NPLG
      COMMON  K1,K2,K3,K9,L1,L2,L3,L9,LG(10),IBRAN
      IF(LI-1)2,1,2
    1 II=1
      ID=0
      IF(LMC(KL)) 5,4,4
    5 ID=1
      LMC(KL)=-LMC(KL)
      GO TO 4
    2 IF(LI-4) 9,3,9
    3 II=2
    4 LT1(II)=L1
      LT2(II)=L2
      LT3(II)=L3
      LT4(II)=L4
      LT9(II)=L9
      KT1(II)=K1
      KT2(II)=K2
      KT3(II)=K3
      KT4(II)=K4
      KT9(II)=K9
      MLIT(II)=LIT
      MKLT(II)=KLT
      MLFT(II)=LFT
    9 GO TO (10,20,30,40,50,60,70,71),LI
   10 DO 11 II=1,2
        TC(II)=0.0
      OC(II)=0.0
      TCT(II)=0.0
      OCT(II)=0.0
      TCZ(II)=0.0
      OCZ(II)=0.0
      TMZ(II)=0.0
      OMZ(II)=0.0
      TM1(II)=0.0
      OM1(II)=0.0
      TBCZ(II)=0.0
   11 OM(II)=0.0
      TBM(II)=0.0
      TF(II)=0.0
      TCTH(II)=0.0
      OCTH(II)=0.0
      OF(II)=0.0
      TC(1)=TQR
      TC(2)=TQE
      OC(1)=ANGR
      OC(2)=ANGE
      ANGR=0.0
      ANGE=0.0
      TQR=0.0
      TQE=0.0
      LF=LOC(KL)
      GO TO 90
   20 TCZ(1)=TQR
      TCZ(2)=TQE
      OCZ(1)=ANGR
      OCZ(2)=ANGE
      ANGR=1.
      ANGE=0.
      TQR=0.
      TQE=0.
      GO TO 97
```

```
30 TCTH(1)=TQR
   TCTH(2)=TQE
   OCTH(1)=ANGR
   OCTH(2)=ANGE
   IF(LMC(KL)) 31,35,31
31 ANGR=0.0
   ANGE=0.0
   TQR=1.0
   TQE=0.0
   GO TO 97
35 TF(1)=0.0
   TF(2)=0.0
   GO TO 45
40 IF(LMC(KL)) 400,55,400
400 TCT(1)=TQR
   TCT(2)=TQE
   OCT(1)=ANGR
   OCT(2)=ANGE
   OF(1)=0.0
   OF(2)=0.0
41 CALL CSUB(OCTH,OCZ,TEMP1)
   CALL CMPY(TEMP1,OF,TEMP1)
   CALL CSUB(OC,OCZ,TEMP2)
   CALL CSUB(TEMP2,TEMP1,TEMP1)
   CALL CSUB(OCT,OCZ,TEMP2)
   CALL CDIV(TEMP1,TEMP2,TF)
   IF(LI-6) 42,44,42
42 CALL CSUB(TCTH,TCZ,TEMP1)
   CALL CMPY(TEMP1,OF,TEMP1)
   CALL CSUB(TCT,TCZ,TEMP2)
   CALL CMPY(TF,TEMP2,TEMP2)
   CALL CADD(TEMP1,TEMP2,TEMP1)
   CALL CADD(TEMP1,TCZ,TRCZ)
   CALL CADD(TRCZ,TC,TEMP1)
   ANGR=OC(1)
   ANGE=OC(2)
   LF=LMC(KL)
   TQR=TEMP1(1)
   TQE=TEMP1(2)
   GO TO (90,90,90,80,98,97,90),LI
44 ANGR=OF(1)
   ANGE=OF(2)
   TQR=TF(1)
   TQE=TF(2)
   J=LFF(KL)
   LF=LOC(KL)
   J1=LF
   J2=LMC(KL)
   II=1
   GO TO 99
45 CALL CSUB(OCTH,OCZ,TEMP1)
   CALL CSUB(OC,OCZ,TEMP2)
   CALL CDIV(TEMP2,TEMP1,OF)
   ANGR=OF(1)
   ANGE=OF(2)
   TQR=TF(1)
   TQE=TF(2)
   GO TO 97
50 TMZ(1)=TQR
   TMZ(2)=TQE
   OMZ(1)=ANGR
   OMZ(2)=ANGE
   OF(1)=1.0
   OF(2)=0.0
   GO TO 41
55 CALL CSUB(TCTH,TCZ,TEMP1)
   CALL CMPY(TEMP1,OF,TEMP1)
   CALL CADD(TEMP1,TCZ,TBM)
   CALL CADD(TC,TBM,TEMP2)
   TQR=TEMP2(1)
```

```
      TOE=TEMP2(2)
      GO TO 71
   60 TM1(1)=TOR
      TM1(2)=TOE
      OM1(1)=ANGR
      OM1(2)=ANGE
      TEMP1(1)=1.0-OM1(1)
      TEMP1(2)=OM1(2)
      CALL CADD(TEMP1,OMZ,TEMP1)
      CALL CDIV(OMZ,TEMP1,OF)
      GO TO 41
   70 TEMP1(1)=TOR
      TEMP1(2)=TOE
      CALL CADD(TEMP1,TC,TEMP1)
      TOR=TEMP1(1)
      TOE=TEMP1(2)
      GO TO 85
   71 KL=KL+1
      IF(KL-NBR) 72,72,73
   72 LF=LFE(KL)
      LI=1
      GO TO 74
   73 LF=NS+2
      LI=0
   74 IF(IO) 75,92,75
   75 J3=KL-1
      IF(LMC(J3)) 77,76,77
   76 J2=LOC(J3)
      GO TO 78
   77 J2=LMC(J3)
   78 J1=LFF(J3)
      LMC(J3)=-LMC(J3)
      DO 79 IO=J1,J2
   79 WRITE(NW,108) IO,THR(IO),THE(IO),TLR(IO),TLE(IO),TRR(IO),TRE(IO)
      GO TO 91
   80 J1=LFE(KL)
      J2=LOC(KL)
      GO TO 990
   85 LF=LMC(KL)
      GO TO 80
   90 LI=LI+1
      IF(IO) 91,92,91
   91 WRITE(NW,100)
      WRITE(NW,101) (TC(II),OC(II),TCT(II),OCT(II),TCZ(II),OCZ(II),
     1II=1,2)
      WRITE(NW,102)
      WRITE(NW,101)(TMZ(II),OMZ(II),TM1(II),OM1(II),TBCZ(II),OM(II),
     1II=1,2)
      WRITE(NW,103)
      WRITE(NW,101)(TBM(II),TF(II),TCTH(II),OCTH(II),OF(II),II=1,2)
   92 RETURN
   97 J=LFE(KL)
      J2=LOC(KL)
      J1=J
      LF=J2
      II=1
      GO TO 99
   98 J2=LMC(KL)
      J=LOC(KL)
      J1=J
      LF=J2
      II=2
   99 L1=LT1(II)
      L2=LT2(II)
      L3=LT3(II)
      L4=LT4(II)
      L9=LT9(II)
      K1=KT1(II)
      K2=KT2(II)
      K3=KT3(II)
```

```
      K4=KT4(II)
      K9=KT9(II)
      LIT=MLIT(II)
      KLT=MKLT(II)
      LFT=MLFT(II)
  990 IF(IO) 90,90,992
  992 WRITE(NW,126)
      WRITE(NW,701)
      DO 991 IO=J1,J2
  991 WRITE(NW,108) IO,THR(IO),THE(IO),TLR(IO),TLE(IO),TRR(IO),TRE(IO)
      GO TO 90
  108 FORMAT(I7,1X,6E12.5)
  126 FORMAT(/ 4X,3HSTA,4X,19HANGULAR DISP.(RAD.)   ,5X, 8HTORQUE-1,2X,
     1 8H(IN-LBS),6X, 8HTORQUE-2,2X,8H(IN-LBS))
  701 FORMAT( 4X,3HNO.,5X,4HREAL,6X,9HIMAGINARY,5X,4HREAL,6X,9HIMAGINARY
     15X,4HREAL,6X,9HIMAGINARY /I7,1X,6E12.5)
  100 FORMAT(/14X,3HT-C,9X,3HO-C,7X,5HT-C,T,3X,9HTHETA-C,T,4X,
     1 5HT-C,0,3X,9HTHETA-C,0)
  101 FORMAT(5X,6E12.5)
  102 FORMAT(/12X,5HT-M,0,3X,9HTHETA-M,0,9X,5HT-M,I,3X,9HTHETA-M,I,
     19X,3HTBC,5X,7HTHETA-M)
  103 FORMAT(/14X,3HTBM,9X,3HT-F,3X,9HT-C,THETA,1X,11HTHETA-C,TM,,
     15X,7HTHETA-F)
      END
      SUBROUTINE CADD(A,B,C)
C     MEMBERNAME R32CADD
      DIMENSION A(2),B(2),C(2)
      C(1)=A(1)+B(1)
      C(2)=A(2)+B(2)
      RETURN
      END
      SUBROUTINE CDIV(A,B,C)
C
C     MEMBERNAME R32CDIV
C
      DIMENSION A(2),B(2),C(2)
      IF(B(2)) 10,5,10
    5 IF(B(1)) 7,6,7
    6 WRITE(6,101)
      GO TO 14
    7 C7=A(1)/B(1)
      C8=A(2)/B(1)
      GO TO 15
   10 IF(B(1)) 12,11,12
   11 C7=A(2)/B(2)
      C8=-A(1)/B(2)
      GO TO 15
   12 C5=B(2)/B(1)
      C6=B(1)+B(2)*C5
      C7   =(A(1)+C5*A(2))/C6
      C8  =(A(2)-A(1)*C5)/C6
      GO TO 15
   14 WRITE(6,100) (A(I),B(I),I=1,2),C7,C8
   15 C(1) =C7
      C(2)=C8
      RETURN
  100 FORMAT(5X,15HDIAGNOSTIC-CDIV/6E13.5)
  101 FORMAT(5X,/24HCOMPLEX DIVISION BY ZERO)
      END
      SUBROUTINE CDIV2(A,B,C,K,M)
C
C     MEMBERNAME R32CDIV2
C
      DIMENSION A(2,2),B(2,2),C(2),TK(2)
      TK(1)=A(2,K)
      TK(2)=B(2,K)
      IF(M-5) 10,5,10
    5 WRITE(6,100) TK(1),TK(2),C(1),C(2)
   10 CALL CDIV(TK,C,TK)
      A(2,K)=TK(1)
```

```
      B(2,K)=TK(2)
      IF(M-5) 20,15,20
   15 WRITE(6,101) TK(1),TK(2)
   20 RETURN
  100 FORMAT(/14X,3HPFR,9X,3HPFE,7X,5HBRATR,7X,5HBRATE/5X,4E12.5)
  101 FORMAT(/10X,7HRATIO-R,5X,7HRATIO-E/5X,2E12.5)
      END
      SUBROUTINE CMPY(A,B,C)
C
C     MEMBERNAME R32CMPY
C
      DIMENSION A(2),B(2),C(2)
      C1=A(1)*B(1)-A(2)*B(2)
      C2=A(1)*B(2)+A(2)*B(1)
      C(1)=C1
      C(2)=C2
      RETURN
      END
      SUBROUTINE CSUB(A,B,C)
C     MEMBERNAME R32CSUB
      DIMENSION A(2),B(2),C(2)
      C(1)=A(1)-B(1)
      C(2)=A(2)-B(2)
      RETURN
      END
      SUBROUTINE MATIN(A,N1,B,M1,DETER,ID)
C
C     MEMBERNAME R32MATIN
C
      DIMENSION A(7,7),B(7,2),INDEX(7,3)
C
C     GENERAL FORM OF DIMENSION STATEMENT
C
      EQUIVALENCE (IROW,JROW), (ICOLU ,JCOLU ), (AMAX, T, SWAP)
C
C     INITIALIZATION
C
      M=M1
      N=N1
   10 DETER =1.0
   15 DO 20 J=1,N
   20 INDEX(J,3) = 0
   30 DO 550 I=1,N
C
C     SEARCH FOR PIVOT ELEMENT
C
   40 AMAX=0.0
   45 DO 105 J=1,N
      IF(INDEX(J,3)-1) 60, 105, 60
   60 DO 100 K=1,N
      IF(INDEX(K,3)-1) 80, 100, 715
   80 IF (      AMAX -ABS (A(J,K))) 85, 100, 100
   85 IROW=J
   90 ICOLU =K
      AMAX=ABS (A(J,K))
  100 CONTINUE
  105 CONTINUE
      INDEX(ICOLU ,3) = INDEX(ICOLU ,3) +1
  260 INDEX(I,1)=IROW
  270 INDEX(I,2)=ICOLU
C
C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
C
  130 IF (IROW-ICOLU ) 140, 310, 140
  140 DETER =-DETER
  150 DO 200 L=1,N
  160 SWAP=A(IROW,L)
  170 A(IROW,L)=A(ICOLU ,L)
  200 A(ICOLU ,L)=SWAP
      IF(M) 310, 310, 210
  210 DO 250 L=1, M
```

```
  220 SWAP=B(IROW,L)
  230 B(IROW,L)=B(ICOLU ,L)
  250 B(ICOLU ,L)=SWAP
C
C     DIVIDE PIVOT ROW BY PIVOT ELEMENT
C
  310 PIVOT   =A(ICOLU ,ICOLU )
      DETER =DETER *PIVOT
  330 A(ICOLU ,ICOLU )=1.0
  340 DO 350 L=1,N
  350 A(ICOLU ,L)=A(ICOLU ,L)/PIVOT
  355 IF(M) 380, 380, 360
  360 DO 370 L=1,M
  370 B(ICOLU ,L)=B(ICOLU ,L)/PIVOT
C
C     REDUCE NON-PIVOT ROWS
C
  380 DO 550 L1=1,N
  390 IF(L1-ICOLU ) 400, 550, 400
  400 T=A(L1,ICOLU )
  420 A(L1,ICOLU )=0.0
      IF(T)430,550,430
  430 DO 450 L=1,N
  450 A(L1,L)=A(L1,L)-A(ICOLU ,L)*T
  455 IF(M) 550, 550, 460
  460 DO 500 L=1,M
  500 B(L1,L)=B(L1,L)-B(ICOLU ,L)*T
  550 CONTINUE
C
C     INTERCHANGE COLUMNS
C
  600 DO 710 I=1,N
  610 L=N+1-I
  620 IF (INDEX(L,1)-INDEX(L,2)) 630, 710, 630
  630 JROW=INDEX(L,1)
  640 JCOLU =INDEX(L,2)
  650 DO 705 K=1,N
  660 SWAP=A(K,JROW)
  670 A(K,JROW)=A(K,JCOLU )
  700 A(K,JCOLU )=SWAP
  705 CONTINUE
  710 CONTINUE
      DO 730 K = 1,N
      IF(INDEX(K,3) -1) 715,720,715
  720 CONTINUE
  730 CONTINUE
      ID=1
  740 RETURN
  715 ID =2
      WRITE(6,760)
  760 FORMAT (5X,18HMATRIX IS SINGULAR)
      GO TO 740
      END
      FUNCTION PANGF(A,B)
C
C     MEMBERNAME R32PANGF
C
      IF (A)-30,50,30
   30 IF (B) 31,60,31
   31 THET = ATAN(B/A)*57.29577951
      PANGF = THET
      IF (A) 32,32,35
   32 PANGF = THET + 180.0
   33 RETURN
   35 IF (B) 36,33,33
   36 PANGF = THET + 360.0
      GO TO 33
   50 IF (B) 51,52,53
   51 PANGF = 270.0
      GO TO 33
```

```
   52 PANGF = 0.0
      GO TO 33
   53 PANGF = 90.0
      GO TO 33
   60 IF (A) 61,52,52
   61 PANGF = -180.0
      GO TO 33
      END
      SUBROUTINE PLNST
C
C     MEMBERNAME R32PLNST
C
      COMMON TRR(2,2),TRE(2,2),TPR(2,2),TPE(2,2),TFSPR(2,2),TFSPE(2,2),
     1 TFPRR(2,2),TFPRE(2,2),TTPCR(2,2),TTPCE(2,2),TTGRR(2,2),TTGRE(2,2)
      COMMON RIP(200),DST(200),DMS(200),DIN(200),BCB(50),LB(50),DDT(200)
     1,BK(50),LEC(2),RS(2),RW(2),PIP(2),PMS(2),PN(2),SS(2),SW(2),SR(2),
     2 ST(2),LBR(20),LBS(20),CCOM(200),AXY1,AXY2,BXY1,BXY2,TLR(200),
     3 TLE(200),THR(200),THE(200),TRR(200),TRE(200),TRR1(200)
      COMMON A(7,7),B(7,2),LS(20),RP(20),RG(20),SG(20),TRR2(200),
     1 TRE1(200),THE1(200),TLE1(200),TRE2(200),TTFR(20),TTFE(20),
     2 PSP(2),THR1(200),TLR1(200),THR2(200),TLR2(200),THE2(200),
     3 TLE2(200),RL(200),PCP(2),PRP(2)   ,DMP(50)
      COMMON IT, K4,L4,MDIAG,FRQ2,ANGR,ANGF,TQR,TQE ,J,NS ,M1,NW,NR,NPLG
      COMMON  K1,K2,K3,K9,L1,L2,L3,L9,LG(10),IBRAN
      C1=RS(K4)+RW(K4)
      C2=C1+RW(K4)
      FLNL=PN(K4)
      A(1,1)=RW(K4)
      A(1,2)=0.
      A(1,3)= FRQ2*PIP(K4)
      A(1,4)=0.
      A(1,5)=0.
      A(1,6)=0.
      A(1,7)=0.
      A(2,1)=1.0
      A(2,2)=-1.0
      A(2,3)=0.
      A(2,4)=-FRQ2*PMS(K4)*C1
      A(2,5)=0.
      A(2,6)=0.
      A(2,7)=0.
      A(3,1)=0.
      A(3,2)=C1
      A(3,3)=0.
      A(3,4)=0.
      A(3,5)=-1.0
      A(3,6)=0.
      A(3,7)=-FRQ2*PCP(K4)
      A(4,1)=C2
      A(4,2)=0.
      A(4,3)=0.
      A(4,4)=0.
      A(4,5)=0.
      A(4,6)=FRQ2*PRP(K4)*ST(K4)-1.0
      A(4,7)=0.
      A(5,1)=0.
      A(5,2)=SW(K4)
      A(5,3)=0.
      A(5,4)=C1
      A(5,5)=0.
      A(5,6)=0.
      A(5,7)=-(FLNL)*C1
      A(6,1)=0.
      A(6,2)=0.
      A(6,3)=-RW(K4)
      A(6,4)=-C1
      A(6,5)=0.
```

```
      A(6,6)=0.
      A(6,7)=0.
      A(7,1)=SR(K4)
      A(7,2)=0.
      A(7,3)= RW(K4)
      A(7,4) =-C1
      A(7,5)=0.
      A(7,6)=FLNL*C2*ST(K4)
      A(7,7)=0.
      C3 =PSP(K4)*FRQ2
      FSR=(TLR(J)-C3*THR(J))/RS(K4)
      FSI=(TLE(J)-C3*THE(J))/RS(K4)
      B(1,1)=RW(K4)*FSR
      B(1,2)=RW(K4)*FSI
      B(2,1)=-FSR
      B(2,2)=-FSI
      B(3,1)=0.
      B(3,2)=0.
      B(4,1)=0.
      B(4,2)=0.
      B(5,1)=0.
      B(5,2)=0.
      IF(M1-1) 440,440,441
  441 IF(L4-IT) 440,443,440
  440 EE1=0.
      EE2=0.
      EE3=0.
      EE4=0.
      GO TO 442
  443 EE1 =AXY1*FLNL
      EE2=BXY1*FLNL
      EE3=AXY2*FLNL
      EE4 =BXY2*FLNL
  442 B(6,1)=-RS(K4)*FLNL*THR(J)+EE1-SS(K4)*FSR
      B(6,2)=-RS(K4)*FLNL*THE(J)+EE2-SS(K4)*FSI
      B(7,1) =EE3
      B(7,2) =EE4
      CALL MATIN (A,7,B,2,CF9,IDD)
      TQR=B(5,1)
      EER=B(6,1)
      ANGR=B(7,1)
      TQE=B(5,2)
      EEI=B(6,2)
      ANGE=B(7,2)
      TFPRR(M1,K4)=B(1,1)/FLNL
      TFPRE(M1,K4)=B(1,2)/FLNL
      TTPCR(M1,K4)=C1*B(2,1)
      TTPCE(M1,K4)=C1*B(2,2)
C TTPCR= TOTAL TORQUE, PLANET-CARRIER, REAL
C TTPCE= TOTAL TORQUE, PLANET-CARRIER, IMAGINARY
      TPR(M1,K4)=B(3,1) /FLNL
      TPE(M1,K4)=B(3,2) /FLNL
      TBR(M1,K4)=B(4,1) /FLNL
      TBE(M1,K4)=B(4,2) /FLNL
      TTGRR(M1,K4)=EER
      TTGRE(M1,K4)=EEI
      TFSPR(M1,K4)=FSR/FLNL
      TFSPE(M1,K4)=FSI/FLNL
      C3=RS(K4)*FSR
      C4=RS(K4)*FSI
      C5=C2*B(1,1)
      C6=C2*B(1,2)
C TEST DIAGNOSTIC
      IF(MDIAG) 600,608,608
  600 IF(K4-1) 621,620,621
  620 WRITE(NW,703)
  621 WRITE(NW,700)
```

```
      WRITE(NW,701)L4,TBR(M1,K4),TBE(M1,K4),C3,C4,TTPCR(M1,K4),        01270000
     1 TTPCE(M1,K4)                                                    01280000
      WRITE(NW,702)                                                    01290000
      WRITE(NW,701)L4,TPR(M1,K4),TPE(M1,K4),C5,C6,TTGRR(M1,K4),        01300000
     1 TTGRE(M1,K4)                                                    01310000
  608 K4=K4+1                                                          01320000
      IF(K4-NFLG) 332,332,333                                          01330000
  332 L4=LFC(K4)                                                       01340000
      RETURN                                                           01350000
  333 L4=NS+2                                                          01360000
      RETURN                                                           01370000
  700 FORMAT(/4X,3HSTA,1X,24HANG. DISP.-PLANET CENTER,1X,              01380000
     1 22HTOT. TORQUE SUN-PLANET,2X,23HTOT TORQUE PLAN-CARRIER)        01390000
  701 FORMAT( 4X,3HNO.,5X,4HREAL,6X,9HIMAGINARY,5X,4HREAL,6X,9HIMAGINARY01400000
     15X,4HREAL,6X,9HIMAGINARY /I7,1X,6E12.5)                          01410000
  702 FORMAT(/ 4X,3HSTA,2X,22HANG. DISP.-PLANET BODY,2X,               01420000
     1 23HTOT. TORQUE PLANET-RING,2X,22HTOT. TORQUE GRND.-RING)        01430000
  703 FORMAT(/15X,          66HALL TORQUES GIVEN IN INCH-LB. AND ANGUL01440000
     1AR DISPLACEMENTS IN RADIANS)                                     01450000
  705 FORMAT(12X,4HREAL,6X,9HIMAGINARY,5X,4HREAL,6X,9HIMAGINARY,5X,    01460000
     14HREAL,6X,9HIMAGINARY)                                           01470000
      END                                                              01480000
***ERROR***      UN-1

      PROGRAMME WAS EXECUTING LINE     3 IN ROUTINE M/PROG WHEN TERMINATION OCCURRED

COMPILE TIME=       0.22 SEC,EXECUTION TIME=      11.55 SEC,OBJECT CODE=

                       512 BYTES,ARRAY AREA=        80 BYTES,UNUSED=     19408 BYTES
```

APPENDIX E

LISTING OF COMPLEX STRUCTURAL DYNAMIC ANALYSIS COMPUTER PROGRAM USING STIFFNESS METHODS (D-82/C-51)

```
LEVEL 21.7 ( MN 73 )                    OS/360  FORTRAN H             DATE  76.125/16.04.23
        COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                          SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                 C      SOURCE MEMBER NAME     DB2ASLK1                           00010000
                 C                                                                00015000
ISN 0002                SUBROUTINE ASLNK1                                         00020000
ISN 0003                IMPLICIT REAL*8(A-H,O-Z)                                  00025000
ISN 0004                REAL*8 DATA,XNONOD,XNOSAT,XNORET,XNORED,XNOPOP,AXNOD,     00026000
                       1 EE,XMU,XNMASV,XNEIG,XNMASG,SCALE,EVSAVE,XKSAVE,XNSPEC,   00027000
                       2AS,FSM,RSM,DM,UM,WONE,OINV,T                              00028000
ISN 0005                COMMON/LINKA1/NTOTAL,NORET,NTAPE                          00030000
ISN 0006                COMMON /LINKA/ LAP(250),                                  00040000
                       1       XNONOD,XNOSAT,XNORET,XNORED,XNOPOP,AXNOD,EE(3),XMU(3),  00050000
                       2 XNMASV,XNEIG,XNMASG,SCALE,EVSAVE,XKSAVE,XNSPEC,          00060000
                       3 AS,FSM,RSM,DM,UM,WONE,OINV                               00061000
ISN 0007                DIMENSION DATA(30),DUM(4),IOP(4),KK(6),MAP(1850),NMAP(1850),  00070000
                       1 NODE1(2750),NODE2(2750),NONE(999),NTWO(999),OPTION(2),   00080000
                       2 ROW(1850),ROW2(12),SEMP(4),TEMP(4),TEMP1(2750),          00090000
                       3 TEMP2(2750),TEMP3(2750),IWOR(4),WORDS(4),X(999),XBK(12,12),  00100000
                       4Y(999),Z(999),TEMP4(2750),TEMP5(2750),I(6)                00110000
ISN 0008                DIMENSION MAP2(6),MAP1(250)                               00111000
ISN 0009                COMMON/IOO/MAP1,NO4                                       00112000
                 C                                                                00120000
ISN 0010                DATA OPTION/'YES','NO'/                                   00121100
ISN 0011                EQUIVALENCE(DATA(1),XNONOD)                               00121200
ISN 0012                DATA IWOR/6H      ,6H      ,6H      ,6H      /            00122000
                 C                                                                00140000
                 C                                                                00150000
ISN 0013                NSPEC=XNSPEC                                              00151000
ISN 0014                NONOD=XNONOD                                              00160000
ISN 0015                NOSAT=XNOSAT                                              00170000
ISN 0016                NORET=XNORET                                              00180000
ISN 0017                NORED=XNORED                                              00190000
ISN 0018                NOPROP=XNOPOP                                             00200000
ISN 0019                MAXNOD=AXNOD                                              00210000
ISN 0020                NMASV=XNMASV                                              00220000
ISN 0021                NEIG=XNEIG                                                00230000
ISN 0022                NMASG=XNMASG                                              00240000
ISN 0023                NO4=DM+.01                                                00241000
                 C                                                                00250000
ISN 0024                NTOTAL=NORET+NORED                                        00260000
ISN 0025                NN=NTOTAL                                                 00270000
ISN 0026                NODEL=6*NONOD-NORET-NORED                                 00280000
ISN 0027                DO 4629  I=1,4                                            00281000
ISN 0028          4629 IOP(I)=DATA(I+19)                                          00282000
ISN 0029                NOYE=2                                                    00283000
ISN 0030                IF(UM.EQ.1) NOYE=1                                        00284000
                 C                                                                00290000
ISN 0032                CALL LOADER (SKP,2)                                       00300000
ISN 0033           999 FORMAT (1H0,29X, 37H FUSELAGE ANALYSIS PROGRAM-MELAN FUD  /22X,  00310000
                       1     '          DB200  SCIARRA / JACKSON / JOANNE' // )   00320000
ISN 0034                WRITE (6,999)                                             00330000
                 C                                                                00340000
```

```
ISN 0035          WRITE (6,985)NONOD,NOSAT,NORET,NORED,NOPROP,MAXNOD,NODEL,NTOTAL   00350000
                 1               ,NMASV,NEIG,NMASG                                   00360000
ISN 0036      985 FORMAT (9X,15H NO. OF NODES = I6,/ 9X,25H NO. OF STRUCTURAL ELEMEN00370000
                 1          NTS = I6 / 9X,23H NO. OF RETAINED DOFS = I6 / 9X,        00380000
                 2          22H NO. OF REDUCED DOFS = I6 / 9X,16H NO. OF MATERIAL    00390000
                 3          13H PROPERTIES = I6/ 9X,21H MAX. NODE NO. USED = I6 / 9X, 00400000
                 4          22H NO. OF DELETED DOFS = I6 / 9X,20H FULL SIZE OF MATRIX 00410000
                 5          19H BEFORE REDUCTION = I6 / 9X,18H NO. OF VARIATION=I6/ 9X00420000
                 6          ,21H NO. OF EIGENVECTORS=I6/ 9X,20H NO. OF MASS GROUPS=I6)00430000
ISN 0037          WRITE(6,4915) NSPEC                                               00441000
ISN 0038     4915 FORMAT( 10X 'NO. OF SPECIAL ELEMENTS=' I6 )                       00442000
            C                                                                       00450000
ISN 0039      242 FORMAT( 10X '*** PUNCHED OUTPUT IS GENERATED ***' )               00452000
            C                                                                       00460000
ISN 0040          IF (EVSAVE.EQ.1.) WRITE (6,927)                                   00470000
            C                                                                       00480000
ISN 0042      927 FORMAT (9X,                                                       00490000
                 159H ***********************************************************/9X,00500000
                 259H *            THE TAPE ENGINEER HAS PRESERVED IS ----           */9X,00510000
                 359H ***********************************************************/) 00520000
            C                                                                       00530000
            C                                                                       00540000
ISN 0043          WRITE(6,986) (EE(I),XMU(I),I=1,NOPROP)                            00550000
ISN 0044      986 FORMAT( / 18X 20H MATERIAL PROPERTIES /10X 8HMODULUS 19X 7HPOISSON00560000
                 1 6H RATIO (7 6X F15.2, 15X F10.4))                                00570000
ISN 0045          WRITE(6,9676) (OPTION(IOP(I)+1),I=1,3)                            00571000
ISN 0046     9676 FORMAT( / 15X 'INPUT OPTIONS' / 10X 'INPUT', 14X A4 / 10X 'FULL K'00572000
                 1 , 13X A4 / 10X 'FINAL K', 12X A4 /)                              00573000
ISN 0047          WRITE(6,9677) WONE                                                00575000
ISN 0048     9677 FORMAT ( 10X 'EIGENVECTORS CALCULATED FOR EIGENVALUES GREATER THAN00576000
                 1', E13.6 )                                                        00577000
            C                                                                       00580000
            C                           READ IN ACTION TABLE                        00590000
            C                                                                       00600000
            C                                                                       00610000
ISN 0049          IF(AS.EQ.1.) GO TO 686                                            00611000
ISN 0051          ICOUNT=1                                                          00620000
ISN 0052          KOUNT=0                                                           00630000
ISN 0053          CALL LOADER (SKP,2)                                               00640000
ISN 0054      200 FORMAT  ( 5X 'SCIARRA - JACKSON - JOANNE USA MATRIX',             00650000
                 1            ' METHOD OF STRUCTURAL ANALYSIS' / )                  00660000
ISN 0055          WRITE (6,200)                                                     00670000
            C                                                                       00680000
ISN 0056      987 FORMAT (32X,14H ACTION TABLE /1X 80(1H-)/                         00690000
                 1 3X 'NODE', 7X 'X', 11X 'Y', 9X 'Z', 6X 'NONE', 3X 'NTWO' / )     00700000
ISN 0057          WRITE (6,987)                                                     00720000
            C                                                                       00730000
ISN 0058      686 JJ=1                                                              00740000
ISN 0059          K=1                                                               00750000
            C                                                                       00760000
ISN 0060          REWIND 7                                                          00760100
ISN 0061          WRITE(7,2)                                                        00761100
```

```
ISN 0052         2 FORMAT(' ASSEMBLY FAT',/,'BEGIN BLOCK INPUT',/,'MATERIAL TABLE')  00761200
ISN 0053           WRITE(7,3)(I,EE(I),XMU(I),I=1,3)                                  00761300
ISN 0054         3 FORMAT(1H ,(I1,',',002,/37,70,',F15.2, ',12.5,',F10.4))          00761400
ISN 0055           WRITE(7,4)                                                        00761500
ISN 0056         4 FORMAT(1H ,'NODAL COORDINATES')                                   00761600
ISN 0067           REWIND 10                                                         00762100
ISN 0068           DO 1 I =1,NONOD                                                   00770000
ISN 0069           READ (5,800) NODE,X(NODE),Y(NODE),Z(NODE),NONE(NODE),NTWO(NODE),  00780000
                 1              (DUM(IJ1),IJ1=1,4)                                   00790000
ISN 0070           T(1)=X(NODE)                                                      00790010
ISN 0071           T(2)=Y(NODE)                                                      00790020
ISN 0072           T(3)=Z(NODE)                                                      00790030
ISN 0073           WRITE(7,5)NODE,T(1),T(2),T(3)                                     00790100
ISN 0074         5 FORMAT(1H ,I3,3(',',E11.4))                                       00790300
ISN 0075           IF(AS.EQ.1.) GO TO 687                                            00791000
ISN 0077           IF(ICOUNT.NE.51) GO TO 9                                          00800000
ISN 0079           CALL LOADER (SKP,2)                                               00810000
ISN 0080           WRITE (6,987)                                                     00820000
ISN 0081           ICOUNT=0                                                          00830000
ISN 0082         9 ICOUNT=ICOUNT+1                                                   00840000
ISN 0083       800 FORMAT (I3,3E10.0,2I3,4A 6)                                       00850000
ISN 0084           WRITE (6,998)NODE,X(NODE),Y(NODE),Z(NODE),NONE(NODE),NTWO(NODE),  00860000
                 1             (DUM(IJ1),IJ1=1,4)                                    00870000
ISN 0085       998 FORMAT( 4X I3, 2X 3F11.4, 3X I3, 3X I3, 2X 4A6 )                  00880000
           C                                                                         00890000
           C       UNPACK DOF OF NODE                                                00900000
           C                                                                         00910000
ISN 0086       687 CALL UNPACK(NONE(NODE),NTWO(NODE),KK)                             00920000
ISN 0087           WRITE(10)NODE,KK                                                  00921000
           C                                                                         00930000
ISN 0088           DO 700 L =1,6                                                     00940000
ISN 0089           IF (KK(L).NE.2) GO TO 32                                          00950000
           C                                                                         00960000
ISN 0091           MAP(JJ)=NODE                                                      00970000
ISN 0092           MAP1(JJ)=L                                                        00971000
ISN 0093           JJ=JJ+1                                                           00980000
ISN 0094           GO TO 700                                                         00990000
ISN 0095        32 IF (KK(L).NE.1) GO TO 700                                         01000000
ISN 0097           KKK=NONET+K                                                       01010000
ISN 0098           MAP(KKK)=NODE                                                     01020000
ISN 0099           K=K+1                                                             01030000
ISN 0100       700 CONTINUE                                                          01040000
ISN 0101         1 CONTINUE                                                          01050000
ISN 0102           REWIND 10                                                         01051100
ISN 0103           WRITE(7,90)                                                       01051200
ISN 0104        90 FORMAT(1H ,'CONSTRAINT DATA-IMPOSED DISPLACEMENTS')               01051400
           C                                                                         01060000
ISN 0105           DO 60 I6=1,NONOD                                                  01061100
ISN 0106           READ(10)NODE,KK                                                   01061200
ISN 0107           K2=1                                                              01061210
ISN 0108           DO 61 II=1,6                                                      01061300
ISN 0109           IF(KK(II).EQ.0) GO TO 62                                          01061400
```

```
ISN 0111          GO TO 61                                                   01061500
ISN 0112       62 MAP2(K2)=II                                                01061600
ISN 0113          K2=K2+1                                                    01061700
ISN 0114       61 CONTINUE                                                   01061800
ISN 0115          K3=K2-1                                                    01061900
ISN 0116          IF(K3.EQ.0) GO TO 60                                       01062100
ISN 0118          IF(K3.EQ.1)WRITE(7,501)(NODE,MAP2(I),I=1,K3)               01062200
ISN 0120          IF(K3.EQ.2)WRITE(7,502)(NODE,MAP2(I),I=1,K3)               01062210
ISN 0122          IF(K3.EQ.3)WRITE(7,503)(NODE,MAP2(I),I=1,K3)               01062220
ISN 0124          IF(K3.EQ.4)WRITE(7,504)(NODE,MAP2(I),I=1,K3)               01062230
ISN 0126          IF(K3.EQ.5)WRITE(7,505)(NODE,MAP2(I),I=1,K3)               01062240
ISN 0128          IF(K3.EQ.6)WRITE(7,506)(NODE,MAP2(I),I=1,K3)               01062250
ISN 0130      501 FORMAT(' /1/,0,/2/,0',1(',',I3,'F',I1))                    01062350
ISN 0131      502 FORMAT(' /1/,0,/2/,0',2(',',I3,'F',I1))                    01062360
ISN 0132      503 FORMAT(' /1/,0,/2/,0',3(',',I3,'F',I1))                    01062400
ISN 0133      504 FORMAT(' /1/,0,/2/,0',4(',',I3,'F',I1))                    01062410
ISN 0134      505 FORMAT(' /1/,0,/2/,0',5(',',I3,'F',I1))                    01062420
ISN 0135      506 FORMAT(' /1/,0,/2/,0',6(',',I3,'F',I1))                    01062430
ISN 0136       60 CONTINUE                                                   01062500
ISN 0137          REWIND 10                                                  01062600
ISN 0138          JSUB=JJ-1                                                  01070000
ISN 0139          IF (JSUB  .EQ.NORET) GO TO 747                             01080000
ISN 0141      988 FORMAT (/ 9X,30H RETAINED DOF COUNT SHOULD BE I6, 8H IT WAS I6 /) 01090000
ISN 0142          WRITE (6,988) JSUB,NORET                                   01100000
ISN 0143          CALL EXIT                                                  01110000
            C                                                                01120000
ISN 0144      747 CONTINUE                                                   01130000
ISN 0145          KSB=K-1                                                    01140000
ISN 0146          IF (KSB.EQ.NORED) GO TO 748                                01150000
ISN 0148      991 FORMAT (18X,29H REDUCED DOF COUNT SHOULD BE I6, 8H IT WAS I6/)   01160000
ISN 0149          WRITE (6,991) KSB,NORED                                    01170000
ISN 0150          CALL EXIT                                                  01180000
ISN 0151      748 CONTINUE                                                   01190000
ISN 0152          CALL LOADER (SKP,2)                                        01200000
ISN 0153          WRITE (6,200)                                              01210000
            C                                                                01220000
ISN 0154      989 FORMAT(1X 80(1H=)/ 18X 14H MAP-RETAINED // (3X 16I4))      01230000
ISN 0155          WRITE (6,989)(MAP(I),I=1,NORET )                           01240000
ISN 0156      911 FORMAT (1X  80(1H=))                                       01250000
ISN 0157          WRITE (6,911)                                              01260000
ISN 0158          IF (NORET.EQ.NN) GO TO 150                                 01270000
ISN 0160          K=NORET+1                                                  01280000
ISN 0161      990 FORMAT( 18X 12H MAP-REDUCED //(3X 16I4))                   01290000
ISN 0162          WRITE (6,990)(MAP(I),I=K,NTOTAL)                           01300000
ISN 0163          WRITE (6,911)                                              01310000
ISN 0164      150 IF(AS.EQ.1.) GO TO 688                                     01311000
ISN 0166          CALL LOADER(FLIP,2)                                        01320000
ISN 0167          WRITE (6,200)                                              01330000
ISN 0168      688 NAXIAL=0                                                   01340000
ISN 0169          NSKIN=0                                                    01350000
ISN 0170          NBEAM=0                                                    01360000
            C                                                                01380000
```

```
ISN 0171       6066 NAP=NOSAT+NSPEC                                                     01384000
ISN 0172            I=0                                                                 01384100
ISN 0173            LK=0                                                                01384200
ISN 0174            ITYPE=9                                                             01384300
ISN 0175            DO 11  INDEX=1,NAP                                                  01385000
ISN 0176            IF(INDEX.GT.NSPEC) GO TO 9065                                       01386000
ISN 0178            DO 28  LS=1,4                                                       01386100
ISN 0179         28 WORDS(LS)=TWOR(LS)                                                  01386200
ISN 0180            NM=0                                                                01387000
ISN 0181            MO=0                                                                01388000
ISN 0182            READ(5,5055) NTYP,LN1,LN2,LN3,LN4,MC,(SEMP(L),L=1,4)                01389000
ISN 0183       5055 FORMAT(5I3,I1,4E10.0)                                               01389010
ISN 0184            WRITE(6,2371) NTYP,LN1,LN2,LN3,LN4,MC,(SEMP(L),L=1,4)               01389020
ISN 0185       2371 FORMAT( 1X 80(1H=) / 21X 'SPECIAL ELEMENT' / '  TYPE =',I4,         01389030
                   1 '  LN1 =',I4, '  LN2 =',I4, '  LN3 =', I4, '  LN4 =', I4 /         01389040
                   2 '  PROPERTIES =',I2, 4( ',', E14.5))                               01389050
ISN 0186            IF(NTYP.NE.5) GO TO 3035                                            01389100
ISN 0188        781 N1=LN1                                                              01389200
ISN 0189            N2=LN2                                                              01389300
ISN 0190            N3=LN3                                                              01389400
ISN 0191            IF(NM.EQ.1) GO TO 5631                                              01389500
ISN 0193            NTYPE=2                                                             01389600
ISN 0194            TEMP(1)=SEMP(4)                                                     01389700
ISN 0195            TEMP(2)=0.                                                          01389800
ISN 0196            TEMP(3)=0.                                                          01389900
ISN 0197            NM=1                                                                01389910
ISN 0198            GO TO 7777                                                          01389920
ISN 0199       5631 NTYPE=4                                                             01389930
ISN 0200            DO 6181  J=1,3                                                      01389940
ISN 0201       6181 TEMP(J)=SEMP(J)                                                     01389950
ISN 0202            NM=0                                                                01389960
ISN 0203            GO TO 7777                                                          01389970
ISN 0204       3035 IF(NTYP.NE.6) GO TO 4045                                            01390000
ISN 0206        881 IF(MO.GT.2) GO TO 3611                                              01391000
ISN 0208            NTYPE=3                                                             01392000
ISN 0209            N3=LN4                                                              01393000
ISN 0210            DO 6212  J=1,3                                                      01394000
ISN 0211       6212 TEMP(J)=SEMP(1)                                                     01395000
ISN 0212            IF(MO.EQ.2) GO TO 3711                                              01396000
ISN 0214            N1=LN1                                                              01397000
ISN 0215            N2=LN2                                                              01398000
ISN 0216            MO=2                                                                01399000
ISN 0217            GO TO 7777                                                          01399100
ISN 0218       3711 N1=LN2                                                              01399200
ISN 0219            N2=LN3                                                              01399300
ISN 0220            MO=3                                                                01399400
ISN 0221            GO TO 7777                                                          01399500
ISN 0222       3611 NTYPE=2                                                             01399600
ISN 0223            N3=0                                                                01399700
ISN 0224            TEMP(2)=0.                                                          01399800
ISN 0225            TEMP(3)=0.                                                          01399900
ISN 0226            IF(MO.EQ.4) GO TO 3811                                              01399910
```

```
ISN 0228          N1=LN1                                                          01399920
ISN 0229          N2=LN2                                                          01399930
ISN 0230          TEMP(1)=SEMP(2)                                                 01399940
ISN 0231          MO=4                                                            01399950
ISN 0232          GO TO 7777                                                      01399960
ISN 0233    3811  N1=LN3                                                          01399970
ISN 0234          N2=LN4                                                          01399980
ISN 0235          TEMP(1)=SEMP(3)                                                 01399990
ISN 0236          MO=0                                                            01400000
ISN 0237          GO TO 7777                                                      01401000
ISN 0238    4045  WRITE(6,7227)                                                   01402000
ISN 0239    7227  FORMAT( // 5X 'INCORRECT IDENTIFICATION FOR SPECIAL ELEMENT' )  01403000
ISN 0240          CALL EXIT                                                       01404000
ISN 0241    9065  READ(5,801) NTYPE,N1,N2,N3,MC,(TEMP(J),J=1,3),CF1,CF2,TXY,B     01406000
ISN 0242          IF(LK.EQ.0)WRITE(7,95)                                          01406100
ISN 0244          IF(LK.EQ.0.AND.NDM.GT.0)WRITE(8,95)                             01406200
ISN 0246      95  FORMAT(' STRUCTURAL DATA')                                      01408300
ISN 0247          MB=MC                                                           01408400
ISN 0248          IF(MB.EQ.0)MB=1                                                 01408500
ISN 0250     801  FORMAT (2X,I1,3I3,I1,6E10.0,E7.0)                               01410000
ISN 0251          LK=LK+1                                                         01410100
ISN 0252          IF(NTYPE.EQ.2) GO TO 70                                         01410200
ISN 0254          IF(NTYPE.EQ.0)NTYPE=3                                           01410250
ISN 0256          IF(NTYPE.EQ.4) GO TO 71                                         01410300
ISN 0258          IF(ITYPE.NE.NTYPE)WRITE(7,96)                                   01410400
ISN 0260          IF(ITYPE.NE.NTYPE.AND.NDM.GT.0)WRITE(8,96)                      01410500
ISN 0262      96  FORMAT(' ELEMENT TYPE=201')                                     01410600
ISN 0263          ITYPE=NTYPE                                                     01410700
ISN 0264          T(2)=TEMP(2)                                                    01410710
ISN 0265          T(3)=TEMP(3)                                                    01410720
ISN 0266          T(1)=TEMP(1)                                                    01410730
ISN 0267          IF(T(2).NE.0.) GO TO 5000                                       01410731
ISN 0269          IF(T(3).NE.0.) GO TO 5000                                       01410732
ISN 0271          T(2)=T(1)                                                       01410733
ISN 0272          T(3)=T(1)                                                       01410734
ISN 0273    5000  T(4)=CF1                                                        01410740
ISN 0274          T(5)=CF2                                                        01410750
ISN 0275          T(6)=TXY                                                        01410760
ISN 0276          WRITE(7,97)LK,N1,N2,N3,T(2),T(3),T(1),B,T(4),T(5),T(6),MB       01410800
ISN 0277          IF(NDM.GT.0)WRITE(8,97)LK,N1,N2,N3,T(2),T(3),T(1),              01411000
                 1B,T(4),T(5),T(6),MB                                             01411100
ISN 0279      97  FORMAT(1H ,I5,3(',',I4),3(',',E10.5),6(',0'),',   ++',/,F4.1,   01411200
                 13(',',E10.5),',M(',I1,')')                                      01411300
ISN 0280          GO TO 8777                                                      01411400
ISN 0281      70  CONTINUE                                                        01411500
ISN 0282          IF(ITYPE.NE.NTYPE) WRITE(7,98)                                  01411600
ISN 0284          IF(ITYPE.NE.NTYPE.AND.NDM.GT.0)WRITE(8,98)                      01411700
ISN 0286      98  FORMAT(' ELEMENT TYPE=101')                                     01411800
ISN 0287          ITYPE=NTYPE                                                     01411900
ISN 0288          T(1)=TEMP(1)                                                    01411910
ISN 0289          T(2)=CF1                                                        01411920
ISN 0290          WRITE(7,99)LK,N1,N2,T(1),T(2),MB                                01412100
```

```
ISN 0291          IF(NDM.GT.0)WRITE(8,99)LK,N1,N2,T(1),T(2),MB                   01412200
ISN 0293       99 FORMAT(1H ,I5,2(',',I4),2(',',E10.5),',M(',I1,')')             01412300
ISN 0294          GO TO 8777                                                      01412400
ISN 0295       71 CONTINUE                                                        01412500
ISN 0296          IF(ITYPE.NE.NTYPE) WRITE(7,63)                                  01412600
ISN 0298          IF(ITYPE.NE.NTYPE.AND.NDM.GT.0)WRITE(8,63)                      01412700
ISN 0300       63 FORMAT(' ELEMENT TYPE=122')                                     01412800
ISN 0301          ITYPE=NTYPE                                                     01412900
ISN 0302          T(2)=TEMP(2)                                                    01412910
ISN 0303          T(1)=TEMP(1)                                                    01412920
ISN 0304          T(3)=TEMP(3)                                                    01412930
ISN 0305          T(4)=CF1                                                        01412940
ISN 0306          WRITE(7,64)LK,N1,N2,N3,T(2),T(1),T(3),T(4),MB                   01413100
ISN 0307          IF(NDM.GT.0)WRITE(8,64)LK,N1,N2,N3,T(2),T(1),T(3),T(4),MB       01413200
ISN 0309       64 FORMAT(1H ,I5,3(',',I4),',0',3(',',E10.5),2(',0'),',',E10.5,    01413300
                 1',M(',I1,')')                                                   01413400
ISN 0310          GO TO 8777                                                      01413500
            C                                                                     01420000
            C                DATA  EXPANSION                                      01430000
            C                                                                     01440000
            C                                                                     01510000
ISN 0311     7777 NOSAT=NOSAT+1                                                   01511000
ISN 0312     8777 I=I+1                                                           01512000
ISN 0313          IF(NTYPE.EQ.0) NTYPE=3                                          01520000
ISN 0315          IF (MC.EQ.0)  MC=1                                              01530000
            C                                                                     01540000
ISN 0317          IF ((NTYPE.NE.3).    (TEMP(2).NE.0.)) GO TO 100                 01550000
ISN 0319          TEMP(2)=TEMP(1)                                                 01560000
ISN 0320          TEMP(3)=TEMP(1)                                                 01570000
ISN 0321      100 CONTINUE                                                        01580000
            C                                                                     01590000
            C                TEST FOR 3 TYPES OF OUTPUT AND PACKING               01600000
            C                                                                     01610000
ISN 0322          IF (NTYPE.NE.2) GO TO 101                                       01620000
            C                                                                     01630000
            C                          PACKING FOR AXIAL ELEMENT                  01640000
            C                                                                     01650000
ISN 0324          IF(AS.EQ.1.) GO TO 681                                          01651000
ISN 0326          WRITE(6,992) N1,N2,MC,TEMP(1),CF1,CF2                           01660000
ISN 0327      992 FORMAT (1X80(1H-)/21X14H AXIAL ELEMENT/2X3HN1=I4,2X3HN2=I4,2X,  01670000
                 2          14HMATERIAL CODE=I2,2X,5HAREA=E12.5,2X,2E12.5 )       01680000
ISN 0328      681 NAXIAL=NAXIAL+1                                                 01690000
ISN 0329          NODE1(I)=N1*100+10*NTYPE+MC                                     01700000
ISN 0330          NODE2(I)=N2*1000+N3                                             01710000
ISN 0331          TEMP1(I)=TEMP(1)                                                01720000
ISN 0332          TEMP2(I)=TEMP(2)                                                01730000
ISN 0333          TEMP3(I)=TEMP(3)                                                01740000
ISN 0334          TEMP4(I)=CF1                                                    01743000
ISN 0335          TEMP5(I)=CF2                                                    01746000
ISN 0336          GO TO 10                                                        01750000
            C                                                                     01760000
ISN 0337      101 IF (NTYPE.NE.3) GO TO 102                                       01770000
```

```
                C                                                                     01780000
                C                       PACKING FOR TRIANGULAR SKIN ELEMENT           01790000
                C                                                                     01800000
ISN 0339              IF(AS.EQ.1.) GO TO 682                                          01801000
ISN 0341              WRITE (6,993)                                                   01810000
ISN 0342          993 FORMAT (1X 80(1H=)/24X 25H TRIANGULAR SKIN ELEMENT  )           01820000
ISN 0343              WRITE(6,994)N1,N2,N3,MC,(TEMP(J),J=1,3),CF1,CF2,T,Y,B           01830000
ISN 0344          994 FORMAT ( 2X, 3HN1=I4, 2X,                                       01840000
                     1         3HN2=I4,2X,3HN3=I4,2X,14HMATERIAL CODE=I2,2X,3HTS=F7.4,2X,01850000
                     23HTX=,F7.4,/,2X,3HTY=,F7.4,2X,4E12.5)                           01860000
ISN 0345          682 NSKIN=NSKIN+1                                                   01870000
                C                                                                     01880000
ISN 0346              NODE1(I)=N1*100+10*NTYPE+MC                                     01890000
ISN 0347              NODE2(I)=N2*1000+N3                                             01900000
ISN 0348              TEMP1(I)=TEMP(1)                                                01910000
ISN 0349              TEMP2(I)=TEMP(2)                                                01920000
ISN 0350              TEMP3(I)=TEMP(3)                                                01930000
ISN 0351              GO TO 10                                                        01940000
                C                                                                     01950000
ISN 0352          102 IF (NTYPE.NE.4) GO TO 103                                       01960000
                C                                                                     01970000
                C                       PACKING FOR BEAM ELEMENT                      01980000
                C                                                                     01990000
ISN 0354              IF(AS.EQ.1.) GO TO 683                                          01991000
ISN 0356              WRITE (6,995)                                                   02000000
ISN 0357          995 FORMAT (1X 80(1H-)/ 21X 14H BEAM ELEMENT )                      02010000
ISN 0358          954 FORMAT ( 2X, 3HN1=I4, 2X,                                       02020000
                     1        3HN2=I4,2X,3HN3=I4,2X,14HMATERIAL CODE=I2,2X,3HIY=E12.5,2X02030000
                     2    ,3HJX=E12.5/2X3HAA=E12.5,2X,2E12.5)                         02040000
ISN 0359              WRITE(6,954) N1,N2,N3,MC,(TEMP(J),J=1,3),CF1,CF2                02050000
ISN 0360          683 NBEAM=NBEAM+1                                                   02060000
ISN 0361              NODE1(I)=N1*100+10*NTYPE+MC                                     02070000
ISN 0362              NODE2(I)=N2*1000+N3                                             02080000
ISN 0363              TEMP1(I)=TEMP(1)                                                02090000
ISN 0364              TEMP2(I)=TEMP(2)                                                02100000
ISN 0365              TEMP3(I)=TEMP(3)                                                02110000
ISN 0366              TEMP4(I)=CF1                                                    02113000
ISN 0367              TEMP5(I)=CF2                                                    02115000
ISN 0368              GO TO 10                                                        02120000
                C                                                                     02130000
ISN 0369          103 WRITE (6,996)NTYPE                                              02140000
ISN 0370          996 FORMAT (//9X,41H TYPE MUST BE 0,2,3,OR4 IN COLM.3 IT WAS I2/)   02150000
ISN 0371              CALL EXIT                                                       02160000
                C                                                                     02170000
ISN 0372           10 IF(INDEX.GT.NSPEC) GO TO 11                                     02180000
ISN 0374              IF(NM.EQ.1) GO TO 781                                           02181000
ISN 0376              IF(MO.NE.0) GO TO 881                                           02182000
ISN 0378           11 CONTINUE                                                        02183000
ISN 0379              WRITE (6,911)                                                   02190000
ISN 0380              NTOT=NAXIAL+NSKIN+NBEAM                                         02200000
ISN 0381              IF (NTOT.EQ.NOSAT) GO TO 50                                     02210000
ISN 0383              WRITE (6,997) NTOT,NOSAT                                        02220000
```

```
ISN 0384      997 FORMAT (9X,                                                            02230000
                  1 61H          ******************   WARNING   ******************/ 02240000
                  2        19X,1H*39H  NO. OF STRUCTURAL ELEMENTS SHOULD BE I6,  1H*/ 02250000
                  3        19X,1H* 9H  IT WAS I6,31X,1H*                          /9X,02260000
                  4 61H          ***************************************************** )02270000
ISN 0385           CALL EXIT                                                          02280000
ISN 0386        30 CONTINUE                                                           02290000
ISN 0387           WRITE (6,977) NIOT,NAXIAL,NSKIN,NBEAM                              02300000
ISN 0388       977 FORMAT (9X,37H TOTAL NUMBER OF STRUCTURAL ELEMENTS I6/9X,           02310000
                  1            23H NO. OF AXIAL ELEMENTS I6 /9X,                      02320000
                  2            23H NO. OF SKIN ELEMENTS  I6/9X,                       02330000
                  3            23H NO. OF BEAM ELEMENTS  I6/)                         02340000
             C                                                                        02350000
ISN 0389           REWIND 1                                                           02350100
ISN 0390           IF(FSM.EQ.1.) GO TO 684                                            02351000
ISN 0392           CALL LOADER (SKP,2)                                                02360000
ISN 0393           WRITE (6,200)                                                      02370000
ISN 0394       964 FORMAT (20X,21HFULL STIFFNESS MATRIX//)                            02380000
ISN 0395           WRITE (6,964)                                                      02390000
             C                                                                        02400000
ISN 0396       684 DO 40  ILL=1,NTOTAL                                                02410000
             C                                                                        02420000
ISN 0397           DO 41 LIL=1,NN                                                     02430000
ISN 0398        41 ROW(LIL)=0.0                                                       02440000
ISN 0399           DO 442 LI2=1,I2                                                    02450000
ISN 0400       442 ROW2(LI2)=0.0                                                      02460000
             C                                                                        02470000
ISN 0401           JJ=MAP(NN)                                                         02480000
             C                                                                        02490000
             C                           LOOK-UP FOR JJ IN SAT                        02500000
             C                                                                        02510000
ISN 0402           DO 42 J =1,NOSAT                                                   02520000
             C                                                                        02530000
ISN 0403           N1=NODE1(J)/100                                                    02540000
ISN 0404           N2=NODE2(J)/1000                                                   02550000
ISN 0405           N3=NODE2(J)-1000*N2                                                02560000
ISN 0406           NOI=N1                                                             02570000
             C                                                                        02580000
ISN 0407           NTYPE=(NODE1(J)-100*NOI)/10                                        02590000
             C                                                                        02600000
ISN 0408           IF (N1.EQ.JJ) GO TO 45                                             02610000
             C                                                                        02620000
ISN 0410           IF (N2.NE.JJ) GO TO 44                                             02630000
ISN 0412           IF (NTYPE.EQ.3) GO TO 45                                           02640000
             C                                                                        02650000
ISN 0414           MAC=N1                                                             02660000
ISN 0415           N1=N2                                                              02670000
ISN 0416           N2=MAC                                                             02680000
ISN 0417           GO TO 45                                                           02690000
             C                                                                        02700000
ISN 0418        44 IF (NTYPE.NE.3) GO TO 42                                           02710000
ISN 0420           IF (N3.NE.JJ) GO TO 42                                             02720000
```

```
              C     ----------------------------------------------------------------02730000
              C                  . UNPACKING FOR AXIAL,SKIN AND BEAM ELEMENTS        02740000
              C                                                                      02750000
ISN 0422         45 MC=NODEI(J)*100+NO1*10+NTYPE                                     02760000
ISN 0423            E=EL(MC)                                                         02770000
ISN 0424            TEMP(1)=XMU(MC)                                                  02780000
ISN 0425            IF (NTYPE.EQ.2) TEMP(I)=TEMP1(J)                                 02790000
ISN 0427            TEMP(2)=TEMP1(J)                                                 02800000
ISN 0428            TEMP(3)=TEMP2(J)                                                 02810000
ISN 0429            TEMP(4)=TEMP3(J)                                                 02820000
ISN 0430            CF1=TEMP4(J)                                                     02823000
ISN 0431            CF2=TEMP5(J)                                                     02825000
              C     ----------------------------------------------------------------02830000
              C                        DETERMINE STRUCTURAL ELEMENTS FOR STRING,SKIN02840000
              C                        AND BEAM, MATRIX (XKB)                        02850000
              C     ----------------------------------------------------------------02860000
ISN 0432            CALL SETUP(NTYPE,N1,N2,N3,E,TEMP,X,Y,Z,NONE,NTWO,XBK,IZ,CF1,CF2) 02870000
ISN 0433            IF (NTYPE.EQ.4) CALL RESORT (XBK)                                02880000
              C     ----------------------------------------------------------------02890000
ISN 0435            IF (NTYPE.EQ.3) CALL SHIFT (N1,N2,N3,JJ,XBK)                     02900000
              C                                                                      02910000
ISN 0437            NBOB=2                                                           02920000
ISN 0438            IF (NN.GT.NORET)  NBOB =1                                        02930000
              C                                                                      02940000
              C     ------------------ SEARING DOF NO. FOR RETAINED, ---------------02950000
              C                        REDUCED OR DELETE                             02960000
ISN 0440            CALL UNPACK (NONE(JJ),NTWO(JJ),KK)                               02970000
ISN 0441            DO 48  I =1,6                                                    02980000
ISN 0442            JK=7-I                                                           02990000
ISN 0443            IF (KK(JK).EQ.NBOB) GO TO 49                                     03000000
ISN 0445         48 CONTINUE                                                         03010000
ISN 0446            GO TO 500                                                        03020000
              C                                                                      03030000
              C     ----------------------------------------------------------------03040000
ISN 0447         49 DO 50 I =1,12                                                    03050000
ISN 0448         50 ROW2(I)=XBK(JK,I)                                                03060000
ISN 0449            IF ((JK.GT.3).AND.(NTYPE.NE.4)) GO TO 42                         03070000
              C                                                                      03080000
              C                PURPOSE OF TRAFER IS 1,UNPACK DOF NO.                 03090000
              C     ---------------------------------- 2,FIND NODE IN MAP(RET,OR,RED) 03100000
              C                                     3,TRANSFER FROM ROW2 TO ROW(NN)03110000
              C                                                                      03120000
              C     ----------------------------------------------------------------03130000
ISN 0451            JO=0                                                             03140000
ISN 0452            CALL TRAFER (N1  ,MAP,NORET,NTOTAL,JO,NONE,NTWO,ROW,ROW2,NTYPE )  03150000
ISN 0453            JO=3                                                             03160000
ISN 0454            IF (NTYPE.EQ.4) JO=6                                             03170000
              C                                                                      03180000
ISN 0456            CALL TRAFER (N2,MAP,NORET,NTOTAL,JO,NONE,NTWO,ROW,ROW2  ,NTYPE )  03190000
              C                                                                      03200000
ISN 0457            IF (NTYPE.NE.3) GO TO 42                                         03210000
ISN 0459            JO=6                                                             03220000
```

```
ISN 0460               CALL TRAPER (NS,MAP,NORET,NTOTAL,JJ,NONE,NTWO,ROW,ROW2,NTYPE )      03230000
                 C                                                                           03240000
ISN 0461            42 CONTINUE                                                              03250000
                 C                                                                           03260000
ISN 0462               IF(FSM.EQ.1.) GO TO 685                                               03261000
ISN 0464           960 FORMAT (.X 80(1H-) / 20X 6H NODE=I4, 1H, 2X 4HROW=I4)                 03270000
ISN 0465               WRITE (6,960) JJ,NN                                                   03280000
ISN 0466               I1=1                                                                  03290000
ISN 0467               I2=I1+3                                                               03300000
ISN 0468               N9=NN-4                                                               03310000
ISN 0469               IF(N9.LE.0) GO TO 553                                                 03320000
ISN 0471               DO 343 IY=1,N9,4                                                      03330000
ISN 0472               DO 342 I3=I1,I2                                                       03340000
ISN 0473           342 IF(ROW(I3).NE.0.) GO TO 331                                           03350000
ISN 0475               GO TO 344                                                             03360000
ISN 0476           331 WRITE (6,962) (I9,ROW(I9),I9=I1,I2)                                   03370000
ISN 0477           962 FORMAT (1X 4(1H(I4,2H)=E13.6) )                                       03371000
ISN 0478           344 I1=I1+4                                                               03391000
ISN 0479           343 I2=I1+3                                                               03400000
ISN 0480           553 WRITE (6,962) (I9,ROW(I9),I9=I1,NN)                                   03410000
                 C                                                                           03420000
ISN 0481           685 IF(NORET.NE.NTOTAL) GO TO 300                                         03430000
ISN 0483               WRITE(1) (ROW(I5),I5=1,NN)                                            03440000
ISN 0484               GO TO 3333                                                            03450000
                 C                                                                           03460000
ISN 0485           300 CONTINUE                                                              03470000
ISN 0486               IA=1                                                                  03480000
ISN 0487               CALL ZOUT(ROW,IA,NN,NMAP)                                             03490000
                 C                                                                           03500000
ISN 0488               LN=NMAP(NN)                                                           03510000
                 C                                                                           03520000
ISN 0489               WRITE (1) (ROW(I5),I5=1,LN)                                           03530000
                 C                                                                           03540000
ISN 0490          3333 CONTINUE                                                              03550000
                 C                                                                           03560000
ISN 0491               KILL=1000*NONE(JJ)+NTWO(JJ)                                           03570000
ISN 0492               KILL=KILL-(10**(6-JK))*NBOB                                           03580000
                 C                                                                           03590000
ISN 0493               NONE(JJ)=KILL/1000                                                    03600000
ISN 0494               NTWO(JJ)=KILL-1000*NONE(JJ)                                           03610000
                 C                                                                           03620000
ISN 0495            40 NN=NN-1                                                               03630000
                 C                                                                           03640000
ISN 0496               REWIND 3                                                              03650000
ISN 0497               WRITE (3) (NMAP(JTK),JTK=1,NTOTAL)                                    03660000
ISN 0498               REWIND 3                                                              03670000
ISN 0499               REWIND 1                                                              03680000
ISN 0500               DO 20 ILAP =1,NORET                                                   03690000
ISN 0501            20 LAP(ILAP)=MAP(ILAP)                                                   03700000
ISN 0502           500 RETURN                                                                03710000
ISN 0503               END                                                                   03720000
```

```
LEVEL 21.7   JAN 73 )                              OS/360  FORTRAN H

         COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=50,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                   C                                                                  00010000
                   C               D82 PROGRAM  -  ALTERNATIVE VERSION OF D4636 USING  00020000
                   C               ROUTINES EIGEN & GELS FROM THE SCIENTIFIC SUBROUTINE 00030000
                   C               PACKAGE (SSP). LARGER RETAINED MATRICES CAN BE     00040000
                   C               HANDLED BY USING THE STORAGE COMPRESSION MODE FOR   00050000
                   C               SYMMETRIC MATRICES.                                00060000
                   C               IF DESIRED, AN ENTRY TO A CSI TYPE OF PARAMETRIC   00070000
                   C               STUDY MAY BE MADE. ONE HAS THE OPTION OF PROVIDING 00080000
                   C               ONE'S OWN MODAL MATRIX OR OF PERFORMING THE        00090000
                   C               D82 CALCULATIONS.                                  00100000
                   C                                                                  00110000
                   C                     MEMBER-NAME  D82MAIN                         00120000
                   C                ADDED TO SM1 ON 10TH, MARCH, 1969                 00130000
                   C                                                                  00140000
  ISN 0002               DOUBLE PRECISION GUL                                         00145000
  ISN 0003               COMMON  /LINKA/  LAP,XNONOD,XNOSAT,XNORET,XNOKED,XNOPOP,     00150000
                        1                 AXNOD,EE,XMU,XNMASV,XNEIG,XNMASS,SCALE,     00160000
                        2                 EVSAVE,XKSAVE,XNSPEC,AS,FSM,KSM,UM,UH,      00170000
                        3                 WONE,UIGV,DUM(3),BIL,BHK,RONKM              00180000
                   C                                                                  00190000
  ISN 0004               COMMON  /LINKA1/   NTOTAL,NORET,NTAPE                        00200000
                   C                                                                  00210000
  ISN 0005               COMMON  /VJT/   GUL                                          00220000
  ISN 0006               COMMON/TKO/IKO,KEY                                           00230000
  ISN 0007               DIMENSION DATA(32),LAP(250),EE(3),GUL(66),XMU(3)             00240000
                   C                                                                  00250000
  ISN 0008               DOUBLE PRECISION  TIME                                       00260000
                   C                                                                  00270000
  ISN 0009               EQUIVALENCE  (DATA(1),XNONOD)                                00280000
                   C                                                                  00290000
                   C                                                                  00300000
  ISN 0010               REWIND 5                                                     00315000
  ISN 0011               IKO=0                                                        00320000
  ISN 0012               KEY=0                                                        00321000
  ISN 0013               CALL ERRSET(208,0,-1,0,0)                                    00325000
  ISN 0014               DO 10 I=1,32                                                 00350000
  ISN 0015            10 DATA(I) = 0.0                                                00360000
  ISN 0016               DO 20 I=1,66                                                 00370000
  ISN 0017            20 GUL(I) = 0.0                                                 00380000
  ISN 0018            30 CALL LOADER(DATA,1)                                          00390000
  ISN 0019               IF (BIL.NE.0.0)  GO TO 50                                    00400000
  ISN 0021               IF(BHK.NE.0.0) EVSAVE=1.0                                    00410000
  ISN 0023               CALL  ASLNK1                                                 00420000
  ISN 0024               CALL  REDLK2                                                 00450000
  ISN 0025               CALL  ABLNK3                                                 00460000
  ISN 0026               M   = XNORET                                                 00510000
  ISN 0027               N   = XNEIG                                                  00520000
  ISN 0028               IF (EVSAVE.NE.1)  GO TO 40                                   00530000
  ISN 0030               CALL  TPCOPY(M,N,3,2)                                        00540000
  ISN 0031            40 CONTINUE                                                     00550000
```

```
ISN 0032            IF (BMK.EQ.0.0)  GO TO 30                            00560000
ISN 0034            IF(BMK.NE.0.) READ(5,100) NINE                       00563000
ISN 0036        100 FORMAT (I5)                                          00566000
ISN 0037         50 CALL  C51LK4                                         00570000
ISN 0038            CALL ABLNKS                                          00571000
ISN 0039            GO TO 30                                             00580000
ISN 0040            END                                                  00590000
```

```
LEVEL 21.7 ( JAN 73 )                              OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                  C                                                                      00010000
                  C              SUBROUTINE  C51LK4  -  ADAPTED FROM C51 PROGRAM          00020000
                  C                                                                      00030000
                  C                          MEMBER-NAME  D82C51                         00040000
                  C                 ADDED TO SM1 ON 7TH, APRIL,1969                      00050000
                  C                                                                      00060000
  ISN 0002              SUBROUTINE  C51LK4                                               00070000
  ISN 0003              IMPLICIT REAL*8(A-H,O-Z)                                         00075000
                  C                                                                      00080000
  ISN 0004              DIMENSION AS(250),C(250),EFM(30,30),EIG(30),                     00090000
                       1FORCE(250),FORCE2(250),G(250),GUL(66),                           00100000
                       2         MAP(250),N1(30),PM(250,30),QI(250,30),                  00110000
                       3SK(30),TMP(250,30),XI(30),XN1(30),SEIG(30)                       00120000
                  C                                                                      00130000
  ISN 0005              COMMON  /GEM/ AS,EIG,FORCE,FORCE2,SK,MAP,NTR,NTC                 00140000
  ISN 0006              COMMON/MMX/EFM,PM,QI,G,NDD3,NSPLIT                               00150000
  ISN 0007              COMMON/VJT/OMG,OLD,FOM,XJP,XN1,XI,DD3TP,SPLIT                    00160000
  ISN 0008              COMMON/TKO/IKO,KEY,N1                                            00161000
                  C                                                                      00170000
  ISN 0009              EQUIVALENCE (GUL(1),OMG),(TMP(1,1),QI(1,1))                      00180000
                  C                                                                      00190000
                  C                                                                      00200000
  ISN 0010              ABS(X)=DABS(X)                                                   00205000
  ISN 0011              DO 10 I=1,250                                                    00210000
  ISN 0012              G(I)  =  0.0                                                     00220000
  ISN 0013              FORCE(I)  = 0.0                                                  00230000
  ISN 0014           10 FORCE2(I)=0.0                                                    00240000
  ISN 0015              DO 15 I=1,250                                                    00253000
  ISN 0016              DO 15 J=1,30                                                     00255000
  ISN 0017           15 QI(I,J)=0.0                                                      00260000
  ISN 0018              CALL DLOADR(GUL,1)                                               00270000
  ISN 0019              NDD3=DD3TP+.1                                                    00275000
  ISN 0020              NTP  =  2                                                        00280000
  ISN 0021          135 REWIND NTP                                                       00290000
  ISN 0022              READ (NTP)  NTR,NTC                                              00300000
  ISN 0023              NSPLIT=SPLIT+.00001                                              00301000
  ISN 0024              NTRIG=0                                                          00302000
  ISN 0025              DO 20 I=1,NTC                                                    00310000
  ISN 0026              SEIG(I)=0.                                                       00311000
  ISN 0027              N1(I)=0                                                          00320000
  ISN 0028              IF(NSPLIT.EQ.0)N1(I)=XN1(I)+.0001                                00320100
  ISN 0030           20 CONTINUE                                                         00320200
  ISN 0031          200 CONTINUE                                                         00321000
  ISN 0032              NC  =  1                                                         00330000
  ISN 0033              I1  =  1                                                         00340000
  ISN 0034              I2  =  1                                                         00350000
  ISN 0035           30 READ (NTP)  (C(I),I=1,NTR)                                       00360000
  ISN 0036              IF (N1(I2).EQ.I1)  GO TO 40                                      00370000
  ISN 0038              I1 = I1 +1                                                       00380000
  ISN 0039              IF (I1.GT.NTC)  GO TO 60                                         00390000
```

```
ISN 0041          GO TO 30                                                      00400000
ISN 0042       40 CONTINUE                                                      00410000
ISN 0043          DO 50 J=1,NTR                                                 00420000
ISN 0044       50 QI(J,NC) = C(J)                                               00430000
ISN 0045          IF(I1.EQ.NTC) GO TO 60                                        00440000
ISN 0047          NC = NC +1                                                    00450000
ISN 0048          I1 = I1 +1                                                    00460000
ISN 0049          I2 = I2 +1                                                    00470000
ISN 0050          GO TO 30                                                      00480000
            C                                                                   00490000
ISN 0051       60 READ (NTP)  MAP                                               00500000
ISN 0052          READ (NTP)  AS                                                00510000
ISN 0053          READ (NTP)  EIG                                               00520000
ISN 0054          DO 65 I=1,NTC                                                 00520300
ISN 0055          J=XNI(I)                                                      00520400
ISN 0056          IF(NTRIG.EQ.1.OR.NTRIG.EQ.2)J=NI(I)                           00520500
ISN 0058          IF(J.NE.0) GO TO 67                                           00520600
ISN 0060          EIG(I)=0.                                                     00520700
ISN 0061          GO TO 65                                                      00520800
ISN 0062       67 EIG(I)=ABS(EIG(J))                                            00520900
ISN 0063          IF(NTRIG.EQ.0)SEIG(I)=EIG(I)                                  00520930
ISN 0065       65 CONTINUE                                                      00520950
ISN 0066          IF(NSPLIT.EQ.0) GO TO 70                                      00521000
ISN 0068          IF(NTRIG.EQ.2) GO TO 63                                       00521100
ISN 0070          IF(NTRIG.EQ.1) GO TO 69                                       00521200
ISN 0072          OMG2=OMG**2                                                   00521300
ISN 0073          IF(OMG2.LT.EIG(1)) GO TO 64                                   00521400
ISN 0075          REWIND NTP                                                    00521500
ISN 0076          READ(NTP)NTR,NTC                                              00521600
ISN 0077          DO 61 I=1,250                                                 00521700
ISN 0078          DO 61 J=1,30                                                  00521800
ISN 0079       61 QI(I,J)=0.                                                    00521900
ISN 0080          NTRIG=1                                                       00522000
ISN 0081          DO 62 I=1,30                                                  00522100
ISN 0082          IF(EIG(I).GT.OMG2) GO TO 200                                  00522200
ISN 0084       62 NI(I)=XNI(I)+.0001                                            00522300
ISN 0085       69 NTRIG=2                                                       00522400
ISN 0086          GO TO 70                                                      00522500
ISN 0087       63 NTRIG=0                                                       00522600
ISN 0088          GO TO 70                                                      00522700
ISN 0089       64 NTRIG=2                                                       00522800
ISN 0090          WRITE(6,301)                                                  00522900
ISN 0091          CALL DLOADR(FORCE,1)                                          00522930
ISN 0092          CALL DLOADR(FORCE2,1)                                         00522940
ISN 0093      301 FORMAT(1H1,10X,'THE FORCING FREQUENCY IS LESS THAN ALL THE EIGENVA00523000
                 1LUES')                                                        00523100
ISN 0094          GO TO 136                                                     00523200
ISN 0095       70 DO 80 I=1,250                                                 00530000
ISN 0096          DO 80 J=1,30                                                  00540000
ISN 0097          PH(I,J) = QI(I,J)                                             00550000
ISN 0098       80 CONTINUE                                                      00560000
ISN 0099          WRITE(10) QI                                                  00565000
```

```
              C                                                                   00570000
ISN 0100            DO 90 I=1,NTR                                                 00580000
ISN 0101            DO 90 J=1,NTC                                                 00590000
ISN 0102         90 TMP(I,J) = AS(I) * PH(I,J)                                    00600000
ISN 0103            DO 110 I=1,NTC                                                00610000
ISN 0104            DO 110 J=1,NTC                                                00620000
ISN 0105            EFM(I,J) = 0.0                                                00630000
ISN 0106            DO 100 K=1,NTR                                                00640000
ISN 0107        100 EFM(I,J) = EFM(I,J) + TMP(K,I)*PH(K,J)                        00650000
ISN 0108        110 CONTINUE                                                      00660000
ISN 0109            DO 120 I=1,NTC                                                00670000
ISN 0110        120 SK(I)=EFM(I,I)*EIG(I)                                         00690000
              C                                                                   00700000
ISN 0111        121 REWIND 10                                                     00704000
ISN 0112            READ (10) QI                                                  00705000
              C                                                                   00710000
ISN 0113        130 CALL  DEE95                                                   00720000
ISN 0114        136 IF(NSPLIT.EQ.0) GO TO 131                                     00721000
ISN 0116            IF(NTRIG.NE.2) GO TO 131                                      00721200
ISN 0118            DO 134 I=1,NTC                                                00721300
ISN 0119            EIG(I)=SEIG(I)                                                00721400
ISN 0120        134 NI(I)=0                                                       00721500
ISN 0121            J=1                                                           00721600
ISN 0122            DO 132 I=1,30                                                 00721700
ISN 0123            IF(XNI(I).EQ.0.) GO TO 133                                    00721800
ISN 0125            IF(EIG(I).LE.OMG2) GO TO 132                                  00721900
ISN 0127            NI(J)=XNI(I)+.00001                                           00722000
ISN 0128            J=J+1                                                         00722100
ISN 0129        132 CONTINUE                                                      00722200
ISN 0130        137 WRITE(6,302)                                                  00722300
ISN 0131            CALL DLOADR(FORCE,1)                                          00722350
ISN 0132            CALL DLOADR(FORCE2,1)                                         00722360
ISN 0133        302 FORMAT(1H1,10X,'THE FORCING FREQUENCY IS GREATER THAN ALL THE EIGE00722400
                   1NVALUES')                                                     00722500
ISN 0134            GO TO 131                                                     00722600
ISN 0135        133 IF(NI(1).EQ.0) GO TO 137                                      00722700
ISN 0137            REWIND NTP                                                    00722720
ISN 0138            READ(NTP)NTR,NTC                                              00722800
ISN 0139            DO 139 I=1,250                                                00722900
ISN 0140            DO 139 J=1,30                                                 00723000
ISN 0141        139 QI(I,J)=0.                                                    00723100
ISN 0142            GO TO 200                                                     00723200
ISN 0143        131 CONTINUE                                                      00723300
ISN 0144            IF(NDO3.EQ.1)NFLG=1                                           00725000
ISN 0146            CALL DLOADR(GUL,1)                                            00730000
ISN 0147            NDO3=DO3TP+.1                                                 00735000
ISN 0148            IF(NFLG.EQ.1) NDO3=0                                          00736000
ISN 0150            JUMP=XJP+.001                                                 00740000
ISN 0151            IF(JUMP.EQ.1) GO TO 135                                       00749000
ISN 0153            IF(JUMP.EQ.2)RETURN                                           00750000
ISN 0155            GO TO 130                                                     00770000
ISN 0156            END                                                           00780000
```

```
LEVEL 21.7 (   N 73 )                               OS/360  FORTRAN H

         COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                C                                                                00010000
                C              THIS ROUTINE DOES THE WORK ORDINARILY             00020000
                C              DONE BY D96.                                      00030000
                C                                                                00040000
                C                            MEMBER-NAME  DB2FRP                 00050000
                C                   ADDED TO SM1 ON 7TH,APRIL,1969               00060000
                C                                                                00070000
  ISN 0002            SUBROUTINE  DEE96                                          00080000
  ISN 0003            IMPLICIT REAL*8(A-H,O-Z)                                   00085000
  ISN 0004            REAL*4 T                                                   00090000
  ISN 0005            DIMENSION AS(250),CI(30),EFC(30),EFF(30),EFM(30,30),       00100000
                    1           EIG(30),FORCE(250),FORCE2(250),G(250),           00110000
                    2           MAP(250),PH(250,30),UI(250,30),RCOS(30),         00120000
                    3           RSIN(30),SK(30),SROOT(30),TMP(250,30),           00130000
                    4           XI(30),XNI(30),YSP(30,30),ZK(30),T(2)            00140000
  ISN 0006            DIMENSION RESLT(215)                                       00145000
  ISN 0007            DIMENSION AAA(215),BBB(215)                                00150000
  ISN 0008            COMMON  /GEM/ AS,EIG,FORCE,FORCE2,SK,MAP,NTR,NTC           00160000
  ISN 0009            COMMON/MMX/EFM,PH,WI,G,NOD3,NSPLIT                         00170000
  ISN 0010            COMMON  /VJT/ OMG,DLO,FOM,XJP,XNI,XI                       00180000
  ISN 0011            COMMON/TOM/MAP1(250),NDM                                   00181000
  ISN 0012            COMMON/TKO/IKO,KEY,NI(30)                                  00190000
  ISN 0013            EQUIVALENCE (YSP(1,1),EFM(1,1)),(TMP(1,1),UI(1,1))         00195000
                C                                                                00200000
  ISN 0014        900 FORMAT  ( 1H1, 35X, 'SCIARRA - JACKSON', / 36X, '( DB2 ',  00210000
                    1           ' PROGRAM )' / 22X, 'DAMPED FORCED RESPONSE ',   00220000
                    2           'OF COMPLEX STRUCTURE' / )                       00230000
  ISN 0015        901 FORMAT (40X,5HINPUT/84(1H-)/)                              00240000
  ISN 0016        902 FORMAT (9X,17H NUMBER OF ROWS =I4 /9X,20H NUMBER OF COLUMNS =I4 / 00250000
                    1           9X, 25H TAPE/CARD INPUT OPTION = F4.1 / )        00260000
  ISN 0017        903 FORMAT(/,18X,' EIGENVECTORS DESIRED',/,(28X,I5))           00270000
  ISN 0018        904 FORMAT (1X,63(1H-)/25X,13H EIGENVALUES  9X,15H MODAL DAMPING 00280000
                    1           /(28X,E12.6,12X,E12.6))                          00290000
  ISN 0019        906 FORMAT (39X,7H MASSES/(9X,5E14.6))                         00300000
  ISN 0020        907 FORMAT (17X,29H FINAL EFFECTIVE MASS MATRIX /64(1H-))      00310000
  ISN 0021        909 FORMAT (29X,14H PSEUDO-VALUES /12X,2H M 14X,2H C 12X,2H K 10X,2H F00320000
                    1           2X,5H SINE 2X,2H F 2X,4H COS /13X,4H EFF 38X,4H EFF  8X, 00330000
                    2           4H EFF  )                                        00340000
  ISN 0022        910 FORMAT (9X,E13.6,2X,E13.6,2X,E13.6,2X,E13.6,2X,E13.6)      00350000
  ISN 0023        911 FORMAT (/32X,21H EXCITING FREQUENCY =F15.5,8H RAD/SEC/84(1H-)) 00360000
  ISN 0024        912 FORMAT (/3X,5H NODE 2X,15HSINE COMPONENT 2X,10HCOSINE COMPONENT 2X00370000
                    1           ,10H RESULTANT  1X,12H PHASE ANGLE 4X,2H G /69X,8H LOADING00380000
                    2           /)                                               00390000
  ISN 0025        913 FORMAT (5X,I3,1X,F15.6,1X,F15.6,  F15.6,1X,F8.2,6X,F9.5  ) 00400000
  ISN 0026        925 FORMAT (34X,13H MODAL MATRIX )                             00410000
  ISN 0027        930 FORMAT (25X,33H OSCILLATORY FORCES FOR THIS CASE /1X,84(1H-)/17X, 00420000
                    1           8H FORCE                                         00430000
                    2           11H COMPONENTS /9X,4H ROW 7X,5H SINE 7X,7H COSINE ) 00440000
  ISN 0028        931 FORMAT (10X,I3,5X,E13.6,2X,E13.6)                          00450000
  ISN 0029        932 FORMAT (29X,14H PSEUDO-VALUES /8X,14H FORCING FREQ,2X,     00460000
```

```
                    1          23H AMPLIFICATION     PHASE 9X,10H AMPLITUDE 726X,          00470000
                    2          23H FACTOR        ANGLE(RAD) )                              00480000
ISN 0030          935 FORMAT (5X,4F15.6)                                                   00490000
ISN 0031          940 FORMAT (30X, 15H SINE COMPONENT )                                    00500000
ISN 0032          941 FORMAT (30X,17H COSINE COMPONENT )                                   00510000
               C                                                                           00520000
               C                                                                           00530000
ISN 0033              ABS(X)=DABS(X)                                                       00531000
ISN 0034              COS(X)=DCOS(X)                                                       00532000
ISN 0035              SIN(X)=DSIN(X)                                                       00533000
ISN 0036              ATN1(X,Y)=DATAN2(X,Y)                                                00534000
ISN 0037              SQRT(X)=DSQRT(X)                                                     00535000
ISN 0038              WRITE (11) EFM,QI                                                    00536000
ISN 0039              DO 10 J=1,NTC                                                        00540000
ISN 0040              CI(J)  = 0.0                                                         00550000
ISN 0041              EIG(J) = ABS(EIG(J))                                                 00560000
ISN 0042              SROOT(J) = SQRT(EIG(J))                                              00570000
ISN 0043           10 CONTINUE                                                             00580000
ISN 0044              NDLO=DLO+.001                                                        00581000
ISN 0045              IF(NSPLIT.EQ.1) GO TO 300                                            00581100
ISN 0047              IF(XJP.NE.1.) GO TO 12                                               00582000
ISN 0049          300 DO 11 J=1,250                                                        00584000
ISN 0050           11 G(J)=0.0                                                             00586000
ISN 0051           12 CONTINUE                                                             00588000
ISN 0052              IF(KEY.GT.0) GO TO 7000                                              00590000
ISN 0054              WRITE(7,67)                                                          00591100
ISN 0055              IF(NUM.GT.0)WRITE(8,67)                                              00591200
ISN 0057           67 FORMAT(' CONSTRAINT DATA-IMPOSED DISPLACEMENTS')                     00591300
ISN 0058         7000 CONTINUE                                                             00591400
ISN 0059              WRITE (6,900)                                                        00600000
ISN 0060              WRITE (6,901)                                                        00610000
ISN 0061              WRITE (6,902)  NTR,NTC,TEST                                          00620000
ISN 0062              WRITE(6,903)(NI(J),J=1,NTC)                                          00630000
ISN 0063              WRITE  (6,904)  (EIG(J),XI(J),J=1,NTC)                               00640000
ISN 0064              WRITE(6,1000)(SROOT(J),J=1,NTC)                                      00641000
ISN 0065         1000 FORMAT(25X,'NATURAL FREQ RAD/SEC',/,(28X,E12.6,/))                   00642000
ISN 0066              WRITE (6,900)                                                        00650000
ISN 0067              WRITE (6,901)                                                        00660000
ISN 0068              DO 30 I=1,NTR                                                        00670000
ISN 0069           30 G(I) = G(I) + AS(I)                                                  00680000
ISN 0070              WRITE (6,906)  (G(J),J=1,NTR)                                        00690000
               C                                                                           00700000
ISN 0071              WRITE (6,900)                                                        00710000
ISN 0072              WRITE (6,925)                                                        00720000
ISN 0073              CALL MATPRN(PH,NTR,NTC,250,30)                                       00730000
ISN 0074              DO 40 I=1,NTR                                                        00740000
ISN 0075              DO 40 J=1,NTC                                                        00750000
ISN 0076           40 TMP(I,J) = 0.0                                                       00760000
               C                                                                           00770000
               C                                  T                                        00780000
               C                       CALCULATE  PHI M PHI                                00790000
               C                                                                           00800000
```

```
ISN 0077          DO 50 I=1,NTR                                               00810000
ISN 0078          DO 50 J=1,NTC                                               00820000
ISN 0079       50 TMP(I,J) = G(I) * PH(I,J)                                   00830000
ISN 0080          DO 70 I=1,NTC                                               00840000
ISN 0081          DO 70 J=1,NTC                                               00850000
ISN 0082          YSP(I,J) = 0.0                                              00860000
ISN 0083          DO 60 K=1,NTR                                               00870000
ISN 0084       60 YSP(I,J)= YSP(I,J) + TMP(K,I)*PH(K,J)                       00880000
ISN 0085       70 CONTINUE                                                    00890000
ISN 0086          WRITE (6,900)                                               00900000
ISN 0087          WRITE (6,907)                                               00910000
ISN 0088          CALL  MATPRN(YSP,NTC,NTC,30,30)                             00920000
            C                                                                 00930000
ISN 0089          DO 75 K=1,250                                               00931000
ISN 0090          FORCE(K)=0.0                                                00932000
ISN 0091       75 FORCE2(K)=0.0                                               00933000
ISN 0092          DO 80 I=1,NTC                                               00940000
ISN 0093          ZK(I) = YSP(I,I)*EIG(I)                                     00950000
ISN 0094    80    CI(I)=YSP(I,I)*SRODT(I)*2.*XI(I)                            00960000
ISN 0095          CALL DLOADR(FORCE,1)                                        00970000
ISN 0096          CALL DLOADR(FORCE2,1)                                       00980000
ISN 0097          WRITE (6,900)                                               00990000
ISN 0098          WRITE (6,930)                                               01000000
ISN 0099          KOUNT = 0                                                   01010000
ISN 0100          DO 110 I=1,NTR                                              01020000
ISN 0101          IF ((FORCE(I).EQ.0.0).AND.(FORCE2(I).EQ.0.0))  GO TO 110    01030000
ISN 0103          IF (KOUNT.LE.45)  GO TO 100                                 01040000
ISN 0105          WRITE (6,900)                                               01050000
ISN 0106          WRITE (6,930)                                               01060000
ISN 0107          KOUNT = 0                                                   01070000
ISN 0108      100 CONTINUE                                                    01080000
ISN 0109          WRITE (6,931) I,FORCE(I),FORCE2(I)                          01090000
ISN 0110          KOUNT = KOUNT +1                                            01100000
ISN 0111      110 CONTINUE                                                    01110000
            C                                                                 01120000
            C                               T                                 01130000
            C              CALCULATE   PHI  * F                               01140000
            C                                  EFF                            01150000
            C                                                                 01160000
ISN 0112          DO 120 I=1,NTC                                              01170000
ISN 0113          EFF(I) = 0.0                                                01180000
ISN 0114          DO 120 K=1,NTR                                              01190000
ISN 0115      120 EFF(I) = EFF(I) + PH(K,I)*FORCE(K)                          01200000
ISN 0116          DO 130 I=1,NTC                                              01210000
ISN 0117          EFC(I) = 0.0                                                01220000
ISN 0118          DO 130 K=1,NTR                                              01230000
ISN 0119      130 EFC(I) = EFC(I) + PH(K,I)*FORCE2(K)                         01240000
ISN 0120          WRITE (6,900)                                               01250000
ISN 0121          WRITE (6,909)                                               01260000
ISN 0122          KOUNT = 0                                                   01270000
ISN 0123          DO 150 I=1,NTC                                              01280000
ISN 0124          IF (KOUNT.LE.45)  GO TO 140                                 01290000
```

```
ISN 0126          WRITE (6,900)                                          01300000
ISN 0127          WRITE (6,909)                                          01310000
ISN 0128          KOUNT = 0                                              01320000
ISN 0129      140 CONTINUE                                               01330000
ISN 0130          KOUNT = KOUNT +1                                       01340000
ISN 0131          WRITE (6,910) YSP(I,I),CI(I),ZK(I),EFF(I),EFC(I)       01350000
ISN 0132      154 CONTINUE                                               01360000
ISN 0133          REWIND 11                                              01364000
ISN 0134          READ(11) EFF,QI                                        01365000
ISN 0135          REWIND 11                                              01365500
C                                                                        01370000
ISN 0136          IF(NDOS.EQ.0) GO TO 175                                01372000
ISN 0138          REWIND 1                                               01373000
ISN 0139          DO 172 J=1,NTC                                         01374000
ISN 0140      172 WRITE(1) (PH(I,J),I=1,NTR)                             01375000
ISN 0141      175 CONTINUE                                               01376000
ISN 0142          OME=OMG                                                01376100
ISN 0143          IF(OMG.EQ.0.)OME=SROOT(NDLO)                           01377000
C                                                                        01380000
ISN 0145      155 WRITE(6,900)                                           01390000
ISN 0146          WRITE (6,911) OME                                      01400000
ISN 0147          WRITE (6,940)                                          01410000
ISN 0148          WRITE (6,932)                                          01420000
ISN 0149          DO 1555 I=1,NTR                                        01421000
ISN 0150          RSIN(I)=0.                                             01422000
ISN 0151     1555 RCOS(I)=0.                                             01423000
ISN 0152          DO 160 I=1,NTC                                         01430000
ISN 0153          IF(SROOT(I).EQ.0.) GO TO 160                           01431000
ISN 0155          BETA = OME/SROOT(I)                                    01440000
ISN 0156          PA = 1.-BETA*BETA                                      01450000
ISN 0157          RB = 2.*XI(I)*BETA                                     01460000
ISN 0158          XMU = 1./SQRT(PA*PA+RB*RB)                             01470000
ISN 0159          PHI = ATAN(RB,PA)                                      01480000
ISN 0160          IF(ZK(I).EQ.0.0) GO TO 156                             01485000
ISN 0162          R = XMU*EFF(I)/ZK(I)                                   01490000
ISN 0163          GO TO 157                                              01493000
ISN 0164      156 R=0.0                                                  01496000
ISN 0165      157 CONTINUE                                               01499000
ISN 0166          RSIN(I) = R*SIN(PHI)                                   01500000
ISN 0167          RCOS(I) = R*COS(PHI)                                   01510000
C                                                                        01520000
C                        WRITE OUT BETA,XMU,PHI,R                        01530000
C                                                                        01540000
ISN 0168          WRITE (6,935)  BETA,XMU,PHI,R                          01550000
ISN 0169      160 CONTINUE                                               01560000
ISN 0170          WRITE (6,900)                                          01570000
ISN 0171          WRITE (6,941)                                          01580000
ISN 0172          WRITE (6,932)                                          01590000
ISN 0173          DO 170 I=1,NTC                                         01600000
ISN 0174          IF(SROOT(I).EQ.0.) GO TO 170                           01601000
ISN 0176          BETA = OME/SROOT(I)                                    01610000
ISN 0177          PA = 1.-BETA*BETA                                      01620000
```

```
ISN 0178          RB = 2.*XT(I)*BETA                                     01630000
ISN 0179          XMU = 1./SQRT(PA*PA+RB*RB)                             01640000
ISN 0180          PHI = ATN1(RB,PA)                                      01650000
ISN 0181          IF(ZK(I).EQ.0.0) GO TO 161                             01655000
ISN 0183          R = XMU*EFC(I)/ZK(I)                                   01660000
ISN 0184          GO TO 162                                              01663000
ISN 0185     161  R=0.0                                                  01666000
ISN 0186     162  CONTINUE                                               01669000
ISN 0187          RSIN(I) = RSIN(I) - R*COS(PHI)                         01670000
ISN 0188          RCOS(I) = RCOS(I) + R*SIN(PHI)                         01680000
ISN 0189          WRITE (6,933)  BETA,XMU,PHI,R                          01690000
ISN 0190     170  CONTINUE                                               01700000
             C                                                           01750000
ISN 0191          KOUNT = 0                                              01760000
ISN 0192          WRITE (6,900)                                          01770000
ISN 0193          WRITE (6,912)                                          01780000
ISN 0194          DO 200 I=1,NTR                                         01790000
ISN 0195          A=0.0                                                  01791000
ISN 0196          B=0.0                                                  01792000
ISN 0197          C=0.0                                                  01793000
ISN 0198          GL=0.0                                                 01794000
ISN 0199          DO 180 J=1,NTC                                         01800000
ISN 0200          A  =  A+ PH(I,J)*RSIN(J)                               01810000
ISN 0201     180  B  =  B+ PH(I,J)*RCOS(J)                               01820000
ISN 0202          PBR=ATN1(A,B)                                          01830000
ISN 0203          IF(PBR.LT.0) PBR=PBR+6.2831852                         01833000
ISN 0205          PBR=PBR*57.29578                                       01836000
ISN 0206          IF (ABS(PBR).GT.360.0) PBR = PBR-360.0                 01840000
ISN 0208          C  = SQRT(A*A+B*B)                                     01850000
ISN 0209          IF(NDD3.EQ.1) RESLT(I)=C                               01855000
ISN 0211          GL = C*OME*OME/386.0                                   01860000
             C                                                           01870000
             C                WRITE OUT AP,SIN,COS,C                     01880000
             C                                                           01890000
ISN 0212          IF (KOUNT.LE.45) GO TO 190                             01900000
ISN 0214          WRITE (6,900)                                          01910000
ISN 0215          WRITE (6,912)                                          01920000
ISN 0216          KOUNT = 0                                              01930000
ISN 0217     190  CONTINUE                                               01940000
ISN 0218          NN = MAP(I)                                            01950000
ISN 0219          NN1=MAP1(I)                                            01951000
ISN 0220          T(1)=A                                                 01951010
ISN 0221          T(2)=B                                                 01951020
ISN 0222          IF(KEY.GT.0) GO TO 69                                  01951030
ISN 0224          WRITE(7,65)T(1),T(2),NN,NN1                            01951100
ISN 0225          IF(NDM.GT.0)WRITE(8,65)T(1),T(2),NN,NN1                01951200
ISN 0227      65  FORMAT(1H ,'/1/,',F14.7,',/2/,',F14.7,',',I5,'F',I1)   01951300
ISN 0228      69  CONTINUE                                               01951400
ISN 0229          WRITE (6,913)  NN,A,B,C,PBR,GL                         01960000
ISN 0230          AAA(I)=A                                               01961000
ISN 0231          BBB(I)=B                                               01962000
ISN 0232     200  KOUNT = KOUNT +1                                       01970000
```

```
ISN 0233         KEY=KEY+9                                                           01970100
ISN 0234         IKU=IKU+1                                                           01970200
ISN 0235         WRITE(12)((AAA(I),I=1,NTR),(BBB(I),I=1,NTR))                        01970300
ISN 0236         IF(NDO3.EQ.1) WRITE(1) (RESLT(J),J=1,NTR)                           01970500
ISN 0238         OME = OME + DLO                                                     01971000
ISN 0239         IF(OME.LE.FOM) GO TO 155                                            01972000
ISN 0241         IF(NDO3.EQ.1) REWIND 1                                              01973000
ISN 0243         WRITE(7,66)                                                         01974000
ISN 0244         IF(NOM.GT.0)WRITE(8,66)                                             01975000
ISN 0246      66 FORMAT(' END BLOCK INPUT',/,'END ASSEMBLY',/,'BEGIN ANALYSIS',/,    01976000
                1'PERFORM A SINGLE ASSEMBLY SAM ANALYSIS',/,'END ANALYSIS')          01977000
ISN 0247         IF(NOM.GT.0)WRITE(8,68)                                             01977100
ISN 0249      68 FORMAT(' END OF ASTRA DECK')                                        01977200
ISN 0250         IF(NOM.GT.0)REWIND 8                                                01978000
ISN 0252         RETURN                                                              01980000
ISN 0253         END                                                                 01990000
```

```
LEVEL 21.7   ( [illegible] 73 )                       OS/360  FORTRAN H

              COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                                 SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                   C      MEMBER NAME S74RESRT
                   C      SOURCE MEMBER NAME      D82RESRT                               00010000
                   C                                                                     00015000
  ISN 0002                SUBROUTINE RESORT (XBK)                                        00020000
  ISN 0003                IMPLICIT REAL*8(A-H,O-Z)                                       00025000
                   C                                                                     00030000
  ISN 0004                DIMENSION XBK(12,12),EXT(12,12),TEMP(12,12)                    00040000
                   C                                                                     00050000
  ISN 0005                DO 1 I =1,12                                                   00060000
  ISN 0006                DO 1 J =1,12                                                   00070000
  ISN 0007              1 EXT(I,J)=0.0                                                   00080000
                   C                                                                     00090000
  ISN 0008                EXT(1,1)=1.                                                    00100000
  ISN 0009                EXT(2,2)=1.                                                    00110000
  ISN 0010                EXT(3,3)=1.                                                    00120000
  ISN 0011                EXT(4,7)=1.                                                    00130000
  ISN 0012                EXT(5,8)=1.                                                    00140000
  ISN 0013                EXT(6,9)=1.                                                    00150000
  ISN 0014                EXT(7,4)=1.                                                    00160000
  ISN 0015                EXT(8,5)=1.                                                    00170000
  ISN 0016                EXT(9,6)=1.                                                    00180000
  ISN 0017                EXT(10,10)=1.                                                  00190000
  ISN 0018                EXT(11,11)=1.                                                  00200000
  ISN 0019                EXT(12,12)=1.                                                  00210000
                   C                                                                     00220000
  ISN 0020                DO 20 I =1,12                                                  00230000
  ISN 0021                DO 20 J =1,12                                                  00240000
  ISN 0022                TEMP(I,J)=0.0                                                  00250000
  ISN 0023                DO 20 K =1,12                                                  00260000
  ISN 0024             20 TEMP(I,J)=TEMP(I,J)+EXT(I,K)*XBK(K,J)                          00270000
                   C                                                                     00280000
  ISN 0025                DO 30 I =1,12                                                  00290000
  ISN 0026                DO 30 J =1,12                                                  00300000
  ISN 0027                XBK(I,J)=0.0                                                   00310000
  ISN 0028                DO 30 K =1,12                                                  00320000
  ISN 0029             30 XBK(I,J)=XBK(I,J)+TEMP(I,K)*EXT(J,K)                           00330000
  ISN 0030                RETURN                                                         00340000
  ISN 0031                END                                                            00350000
```

```
LEVEL 21.7   (JAN 73 )                         OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                 C      MEMBER NAME S74APLDR
                 C      SOURCE MEMBER NAME    DB2LOADR                    00010000
                 C                                                        00010200
  ISN 0002              SUBROUTINE LOADER(ARRAY,MODE)                     00010400
  ISN 0003              REAL*8 DUMY2,DO3TP                                00010500
  ISN 0004              COMMON /LINKA/ DUMY(266),EVSAVE                   00011000
  ISN 0005              COMMON/VJT/DUMY2(64),DO3TP                        00012000
  ISN 0006              DIMENSION  CARD(5),TITLE(19),ARRAY(1)             00020000
  ISN 0007              GO TO (100,200),MODE                              00030000
  ISN 0008       100    CONTINUE                                          00040000
  ISN 0009       105    READ(5,101)NSUB,NOPC,CARD                         00050000
  ISN 0010       101    FORMAT(I4,I1,5E14.7)                              00060000
  ISN 0011              IF(NSUB)303,303,106                               00070000
  ISN 0012       106    IF(NOPC)303,303,107                               00080000
  ISN 0013       107    IF(NOPC-6)109,303,108                             00090000
  ISN 0014       108    IF(NOPC-8)102,111,112                             00100000
  ISN 0015       109    DO 110  JCARD=1,NOPC                              00110000
  ISN 0016              JARRAY=NSUB+JCARD-1                               00120000
  ISN 0017       110    ARRAY(JARRAY)=CARD(JCARD)                         00130000
  ISN 0018              GO TO 105                                         00140000
  ISN 0019       102    READ(5,103)TITLE                                  00150000
  ISN 0020       103    FORMAT(I6,17A4,A2)                                00160000
  ISN 0021              IF(TITLE(1))301,104,301                           00170000
  ISN 0022       104    NPAGE=0                                           00180000
  ISN 0023              GO TO 105                                         00190000
  ISN 0024       111    RETURN                                            00200000
  ISN 0025        112   IF(DO3TP.EQ.1.) STOP                              00210000
  ISN 0027              IF(EVSAVE.EQ.1.) STOP 13                          00210400
  ISN 0029              STOP 14                                           00210500
  ISN 0030       200    NPAGE=NPAGE+1                                     00220000
  ISN 0031              WRITE(6,201)(TITLE(N),N=2,19),NPAGE               00230000
  ISN 0032       201    FORMAT(1H1,17A4,A2,6H PAGE ,I3//)                 00240000
  ISN 0033              RETURN                                            00250000
  ISN 0034       301    WRITE(6,302)                                      00260000
  ISN 0035       302    FORMAT(20H INVALID HEADER CARD)                   00270000
  ISN 0036              STOP                                              00280000
  ISN 0037       303    WRITE(6,304)NSUB,NOPC,ARRAY                       00290000
  ISN 0038       304    FORMAT(19H INVALID DATA CARD ,I4,I1,5E14.7)       00300000
  ISN 0039              STOP                                              00310000
  ISN 0040              END                                               00320000
```

```
            COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                               SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                  C       MEMBER NAME  S74ZOUT                                          00010000
                  C                                                                     00015000
   ISN 0002               SUBROUTINE ZOUT (RO,IA,NN,NMAP)                               00020000
   ISN 0003               IMPLICIT REAL*8(A-H,O-Z)                                      00025000
   ISN 0004               DIMENSION RO(1),NMAP(1)                                       00030000
                  C                                                                     00040000
   ISN 0005               ABS(Q)=DABS(Q)                                                00041000
   ISN 0006               DO 1 I =IA,NN                                                 00050000
                  C                                                                     00060000
   ISN 0007               IF ((ABS(RO(I))-1.E-1).LT.0.) RO(I)=0.0                      00070000
   ISN 0009             1 CONTINUE                                                      00080000
                  C                                                                     00090000
                  C                                                                     00100000
                  C                                                                     00110000
   ISN 0010               I=IA                                                          00120000
   ISN 0011               M=1                                                           00130000
   ISN 0012               MM=0                                                          00140000
                  C                                                                     00150000
   ISN 0013            50 IF (RO(I).NE.0.) GO TO 20                                     00160000
                  C                                                                     00170000
   ISN 0015               MM=MM+1                                                       00180000
                  C                                                                     00190000
   ISN 0016               IF (I.EQ.NN) GO TO 10                                         00200000
                  C                                                                     00210000
   ISN 0018               I=I+1                                                         00220000
   ISN 0019               GO TO 50                                                      00230000
                  C                                                                     00240000
   ISN 0020            10 RO(M)=FLOAT(MM)+1.E-6                                         00250000
   ISN 0021               GO TO 60                                                      00260000
                  C                                                                     00270000
   ISN 0022            20 IF (MM.EQ.0) GO TO 21                                         00280000
                  C                                                                     00290000
   ISN 0024               RO(M)=FLOAT(MM)+1.E-6                                         00300000
   ISN 0025               M=M+1                                                         00310000
   ISN 0026               RO(M)=RO(I)                                                   00320000
   ISN 0027               GO TO 60                                                      00330000
                  C                                                                     00340000
   ISN 0028            21 RO(M)=RO(I)                                                   00350000
                  C                                                                     00360000
   ISN 0029            60 IF (I.EQ.NN) GO TO 70                                         00370000
   ISN 0031               I=I+1                                                         00380000
   ISN 0032               M=M+1                                                         00390000
   ISN 0033               MM=0                                                          00400000
   ISN 0034               GO TO 50                                                      00410000
                  C                                                                     00420000
   ISN 0035            70 NMAP(NN)=M                                                    00430000
   ISN 0036               RETURN                                                        00440000
   ISN 0037               END                                                           00450000
```

```
LEVEL 21.7    AN 73 )                          OS/360  FORTRAN H

           COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                                 SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                   C      MEMBER NAME S74SRCH                                           00010000
                   C                                                                    00015000
  ISN 0002                SUBROUTINE SEARCH (MAP,N1,NORET,NTOTAL,NBOB,II)              00020000
  ISN 0003                IMPLICIT REAL*8(A-H,O-Z)                                      00025000
                   C                                                                    00030000
                   C                     LOOK-UP FOR II(START OF THE NODE IN MAP, RET OR RED)00040000
                   C                                                                    00050000
  ISN 0004                DIMENSION MAP(1500)                                           00060000
  ISN 0005                II=0                                                          00070000
  ISN 0006                IF (NBOB.NE.2) GO TO 3                                        00080000
  ISN 0008                DO 1 I =1,NORET                                               00090000
  ISN 0009                IF (N1.NE.MAP(I)) GO TO 1                                     00100000
  ISN 0011                II=I                                                          00110000
  ISN 0012                GO TO 2                                                       00120000
  ISN 0013              1 CONTINUE                                                      00130000
  ISN 0014                GO TO 2                                                       00140000
  ISN 0015              3 J=NORET+1                                                     00150000
  ISN 0016                IF (NORET.EQ.NTOTAL) GO TO 2                                  00160000
  ISN 0018                DO 4 I =J,NTOTAL                                              00170000
  ISN 0019                IF (N1.NE.MAP(I)) GO TO 4                                     00180000
  ISN 0021                II=I                                                          00190000
  ISN 0022                GO TO 2                                                       00200000
  ISN 0023              4 CONTINUE                                                      00210000
  ISN 0024              2 RETURN                                                        00220000
  ISN 0025                END                                                           00230000
```

```
LEVEL 21.7   (JAN 73 )                          OS/360  FORTRAN H

               COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                                  SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                     C     MEMBER NAME S74TRAFR                                          00010000
                     C                                                                   00015000
  ISN 0002                 SUBROUTINE TRAFER (NODE,MAP,NORET,NTOTAL,JU,NONE,NTWO,ROW,ROW2,   00020000
                         1                       NTYPE  )                                00030000
  ISN 0003                 IMPLICIT REAL*8(A-H,O-Z)                                      00040000
                     C                          TRANSFER VARIABLES                       00050000
                     C                                                                   00060000
  ISN 0004                 DIMENSION MAP(3000),NONE(250),NTWO(250),ROW(3000),ROW2(12),KK(6)   00070000
                     C                                                                   00080000
                     C                          UNPACK DUF NO.                           00090000
                     C                                                                   00100000
  ISN 0005                 CALL UNPACK (NONE(NODE),NTWO(NODE),KK)                        00110000
                     C                                                                   00120000
                     C                          FIND NODE IN MAP                         00130000
                     C                                                                   00140000
  ISN 0006                 DO 5 I =1,2                                                   00150000
                     C                                                                   00160000
                     C                                      TO LOCATE NODE IN MAP WE CALL SEARCH,   00170000
                     C                                      IN ORDER TO GET II (WHICH IS THE START OF THE00180000
                     C                                      NODE IN MAP RET. OR RED. PORTION)   00190000
  ISN 0007                 CALL SEARCH (MAP,NODE,NORET,NTOTAL,I     ,II)                 00200000
  ISN 0008                 IF (II.EQ.0) GO TO 5                                          00210000
                     C                          TRANSFER FROM ROW2 TO ROW                00230000
                     C                                                                   00240000
  ISN 0010                 DO 6 J =1,6                                                   00250000
  ISN 0011                 IF (KK(J).NE.I) GO TO 6                                       00260000
  ISN 0013                 IF ((NTYPE.NE.4).AND.(J.GT.3)) GO TO 5                        00270000
  ISN 0015                 JE=JU+J                                                       00280000
  ISN 0016                 ROW(II)=ROW2(JE)+ROW(II)                                      00290000
  ISN 0017                 II=II+1                                                       00300000
  ISN 0018               6 CONTINUE                                                      00310000
  ISN 0019               5 CONTINUE                                                      00320000
  ISN 0020                 RETURN                                                        00330000
  ISN 0021                 END                                                           00340000
```

```
LEVEL 21.7   (AN 73 )                                OS/360  FORTRAN H

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
              C     MEMBER NAME S74UNPCK                                        00010000
              C                                                                 00015000
 ISN 0002           SUBROUTINE UNPACK (NONE,NTWO,KK)                            00020000
 ISN 0003           IMPLICIT REAL*8(A-H,O-Z)                                    00025000
 ISN 0004           DIMENSION KK(1)                                             00030000
 ISN 0005           KK(1)=NONE/100                                              00040000
 ISN 0006           KK(2)=(NONE-100*KK(1))/10                                   00050000
 ISN 0007           KK(3)=NONE-100*KK(1)-10*KK(2)                               00060000
 ISN 0008           KK(4)=NTWO/100                                              00070000
 ISN 0009           KK(5)=(NTWO-100*KK(4))/10                                   00080000
 ISN 0010           KK(6)=NTWO-100*KK(4)-10*KK(5)                               00090000
 ISN 0011           RETURN                                                      00100000
 ISN 0012           END                                                         00110000
```

```
LEVEL 21.7   'AN 73 )                              OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                    C      MEMBER NAME 974SETUP
                    C
  ISN 0002                 SUBROUTINE SETUP(NTYPE,N1,N2,N3,E,TEMP,X,Y,Z,NONE,NTWO,XBK,IZ,
                         1 CF1,CF2)
  ISN 0003                 IMPLICIT REAL*8(A-H,O-Z)
  ISN 0004                 DIMENSION XBK(12,12),X(250),Y(250),Z(250),NONE(4),NTWO(4),TEMP(4)
  ISN 0005                 DO 166 I =1,12
  ISN 0006                 DO 166 J =1,12
  ISN 0007             166 XBK(I,J)=0.0
                    C
                    C
  ISN 0008                 IF(NTYPE.NE.2) GO TO 24
  ISN 0010                 CALL STRING(X(N1),Y(N1),Z(N1),X(N2),Y(N2),Z(N2),E,TEMP(1),
                         1 XBK,CF1,CF2)
                    C
  ISN 0011                 IZ=6
  ISN 0012                 GO TO 30
                    C
                    C
  ISN 0013              24 IF(NTYPE.NE.3) GO TO  25
  ISN 0015                 G=E/(2.*(1.+TEMP(1)))
  ISN 0016                 CALL SKIN (X(N1),Y(N1),Z(N1),X(N2),Y(N2),Z(N2),X(N3),Y(N3),Z(N3),
                         1             E,G,TEMP(2),TEMP(3),TEMP(4),TEMP(1),XBK)
                    C
  ISN 0017                 IZ=9
  ISN 0018                 GO TO 30
                    C
                    C
  ISN 0019              25 IF(NTYPE.NE.4) GO TO 26
  ISN 0021                 CALL BEAM (X(N1),Y(N1),Z(N1),X(N2),Y(N2),Z(N2),X(N3),Y(N3),Z(N3),
                         1  E,TEMP(1),TEMP(2),TEMP(3),TEMP(4),XBK,CF1,CF2)
  ISN 0022                 IZ=12
  ISN 0023                 GO TO 30
                    C
  ISN 0024              26 WRITE (6,10)
  ISN 0025              10 FORMAT(9X,35HNTYPE NO. IN COLUMN 4 IS NOT 2,3OR4  )
                    C
  ISN 0026              30 CONTINUE
                    C
                    C
                    C
                    C
  ISN 0027                 RETURN
  ISN 0028                 END
```

```
LEVEL 21.7 ( JAN 73 )                              OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                   C      MEMBER NAME  S74BEAM                                        00010000
                   C                                                                  00015000
 ISN 0002                 SUBROUTINE BEAM(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,E,XMU,YI,XJ,AW,BMAT,   00020000
                         1 CF1,CF2)                                                   00021000
 ISN 0003                 IMPLICIT REAL*8(A-H,O-Z)                                    00025000
 ISN 0004                 DIMENSION BMAT(12,12),OME(6,12),BK(6,6),TEMP(6,12),D(3,3)   00030000
                   C                                                                  00040000
 ISN 0005                 DOUBLE PRECISION LI,TI,JI,MU                                00050000
                   C                                                                  00060000
 ISN 0006                 SQRT(X) = DSQRT(X)                                          00061000
 ISN 0007                 ABS(X) = DABS(X)                                            00062000
 ISN 0008                 MU=XMU                                                      00070000
 ISN 0009                 X31=X3-X1                                                   00080000
 ISN 0010                 Y31=Y3-Y1                                                   00100000
 ISN 0011                 Z31=Z3-Z1                                                   00120000
 ISN 0012                 X12=X1-X2                                                   00170000
 ISN 0013                 Y12=Y1-Y2                                                   00180000
 ISN 0014                 Z12=Z1-Z2                                                   00190000
 ISN 0015                 X13=X1-X3                                                   00200000
 ISN 0016                 Y13=Y1-Y3                                                   00210000
 ISN 0017                 Z13=Z1-Z3                                                   00220000
 ISN 0018                 SO1=(1./3.)                                                 00230000
 ISN 0019                 SO3=(2./3.)                                                 00240000
                   C                                                                  00250000
 ISN 0020                 I1=YI                                                       00260000
 ISN 0021                 A=AW                                                        00270000
 ISN 0022                 J1=XJ                                                       00280000
                   C                                                                  00290000
 ISN 0023                 SL=SQRT(X12**2+Y12**2+Z12**2)                               00300000
 ISN 0024                 G=E/(2.*(1.+MU))                                            00310000
 ISN 0025                 SA=0.                                                       00311000
 ISN 0026                 IF(A.EQ.0.) GO TO 105                                       00312000
 ISN 0028                 SA=(12.*E*I1)/(G*A*SL**2)                                   00320000
 ISN 0029             105 F=(6.*E*I1)/(SL**3*(1.+SA))                                 00330000
                   C                                                                  00340000
                   C                                                                  00350000
                   C                                                                  00360000
                   C          FORM 6X6 MATRIX  BK                                     00370000
 ISN 0030                 DO 5 I =1,6                                                 00380000
 ISN 0031                 DO 6 J =1,6                                                 00390000
 ISN 0032               6 BK(I,J)=0.0                                                 00400000
 ISN 0033               5 CONTINUE                                                    00410000
 ISN 0034                 BK(1,1)=2.*F*6.*CF1/(5.*SL)                                 00420000
 ISN 0035                 BK(3,1)=-SL*F-CF1/10.                                       00430000
 ISN 0036                 BK(4,1)=-2.*F                                               00440000
 ISN 0037                 BK(6,1)=-SL*F                                               00450000
                   C                                                                  00460000
 ISN 0038                 BK(2,2)=G*J1/SL                                             00470000
 ISN 0039                 BK(5,2)=-(G*J1)/SL                                          00480000
                   C                                                                  00490000
```

```
ISN 0040          BK(3,3)=SU3*SL**2*F*(1.+SA/4.)+2.*SL*CF1/15.              00500000
ISN 0041          BK(4,3)=SL*F                                               00510000
ISN 0042          BK(6,3)=SU3*SL**2*F*(1.-SA/2.)                             00520000
            C                                                                00530000
ISN 0043          BK(4,4)=2.*F+6.*CF2/(5.*SL)                                00540000
ISN 0044          BK(6,4)=SL*F+CF2/10.                                       00550000
            C                                                                00560000
ISN 0045          BK(5,5)=G*J1/SL                                            00570000
            C                                                                00580000
ISN 0046          BK(6,6)=SU3*SL**2*F*(1.+SA/4.)+2.*SL*CF2/15.              00590000
            C                                                                00600000
ISN 0047          DO  8 I =1,6                                               00610000
ISN 0048          DO  8 J =1,6                                               00620000
ISN 0049        8 BK(I,J)=BK(J,I)                                            00630000
            C                                                                00640000
ISN 0050          BK(4,1)=BK(4,1)-6.*CF1/(5.*SL)                             00641000
ISN 0051          BK(1,4)=BK(1,4)-6.*CF2/(5.*SL)                             00642000
ISN 0052          BK(6,1)=BK(6,1)-CF1/10.                                    00643000
ISN 0053          BK(1,6)=BK(1,6)-CF2/10.                                    00644000
ISN 0054          BK(4,3)=BK(4,3)+CF1/10.                                    00645000
ISN 0055          BK(3,4)=BK(3,4)+CF2/10.                                    00646000
ISN 0056          BK(6,3)=BK(6,3)-SL*CF1/30.                                 00647000
ISN 0057          BK(3,6)=BK(3,6)-SL*CF2/30.                                 00648000
            C                                                                00650000
            C       CALCULATE  DIRECTION COSINES                             00660000
            C                                                                00670000
            C                                                                00680000
ISN 0058          X21=X2-X1                                                  00690000
ISN 0059          Y21=Y2-Y1                                                  00700000
ISN 0060          Z21=Z2-Z1                                                  00710000
            C                                                                00720000
ISN 0061          DX=SQRT(X21**2+Y21**2+Z21**2)                              00730000
ISN 0062          XBX=X21/DX                                                 00740000
ISN 0063          YBX=Y21/DX                                                 00750000
ISN 0064          ZBX=Z21/DX                                                 00760000
            C                                                                00770000
ISN 0065          DO 100 J =1,3                                              00780000
ISN 0066      100 D(J,1)=1.                                                  00790000
            C                                                                00800000
ISN 0067          D(1,2)=Y1                                                  00810000
ISN 0068          D(2,2)=Y2                                                  00820000
ISN 0069          D(3,2)=Y3                                                  00830000
            C                                                                00840000
ISN 0070          D(2,3)=Z2                                                  00850000
ISN 0071          D(1,3)=Z1                                                  00860000
ISN 0072          D(3,3)=Z3                                                  00870000
            C                                                                00880000
ISN 0073          CALL DET (D,A)                                             00890000
            C                                                                00900000
ISN 0074          DO 101 J =1,3                                              00910000
ISN 0075      101 D(J,2)=1.                                                  00920000
            C                                                                00930000
```

```
ISN 0076          D(1,1)=X1
ISN 0077          D(2,1)=X2
ISN 0078          D(3,1)=X3
            C
ISN 0079          CALL DET (D,B)
            C
ISN 0080          DO 102 J =1,3
ISN 0081      102 D(J,3)=1.
            C
ISN 0082          D(1,2)=Y1
ISN 0083          D(2,2)=Y2
ISN 0084          D(3,2)=Y3
            C
ISN 0085          CALL DET (D,C)
            C
ISN 0086          BD=SQRT(A**2+B**2+C**2)
ISN 0087          XBY=A/BD
ISN 0088          YBY=B/BD
ISN 0089          ZBY=C/BD
            C
ISN 0090          XBZ=(YBX*ZBY)-(YBY*ZBX)
ISN 0091          YBZ=(XBY*ZBX)-(XBX*ZBY)
ISN 0092          ZBZ=(XBX*YBY)-(XBY*YBX)
            C
ISN 0093          DO 9 I =1,6
ISN 0094          DO 10 J=1,12
ISN 0095       10 OME(I,J)=0.0
ISN 0096        9 CONTINUE
            C
ISN 0097          OME(1,1)=XBZ
ISN 0098          OME(1,2)=YBZ
ISN 0099          OME(1,3)=ZBZ
            C
ISN 0100          OME(4,4)=XBZ
ISN 0101          OME(4,5)=YBZ
ISN 0102          OME(4,6)=ZBZ
            C
ISN 0103          OME(2,7)=XBX
ISN 0104          OME(3,7)=XBY
ISN 0105          OME(2,8)=YBX
ISN 0106          OME(3,8)=YBY
ISN 0107          OME(2,9)=ZBX
ISN 0108          OME(3,9)=ZBY
            C
ISN 0109          OME(5,10)=XBX
ISN 0110          OME(5,11)=YBX
ISN 0111          OME(5,12)=ZBX
            C
ISN 0112          OME(6,10)=XBY
ISN 0113          OME(6,11)=YBY
ISN 0114          OME(6,12)=ZBY
            C
```

```
              C
ISN 0115            DO 3 I =1,6
ISN 0116            DO 3 J =1,12
ISN 0117            TEMP(I,J)=0.0
              C
ISN 0118            DO 3 K =1,6
ISN 0119          3 TEMP(I,J)=TEMP(I,J)+BK(I,K)*OME(K,J)
              C
ISN 0120            DO 11 I =1,12
ISN 0121            DO 11 J =1,12
ISN 0122            BMAT(I,J)=0.0
ISN 0123            DO 13 K =1,6
              C
ISN 0124         13 BMAT(I,J)=BMAT(I,J)+OME(K,I)*TEMP(K,J)
ISN 0125            IF (ABS(BMAT(I,J)).GT..1) GO TO 11
ISN 0127            BMAT(I,J)=0.0
ISN 0128         11 CONTINUE
              C
ISN 0129            RETURN
ISN 0130            END
```

```
LEVEL 21.7    JAN 73 )                          OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                 C     MEMBER NAME S74SHIFT                                             00010000
                 C                                                                      00015000
   ISN 0002            SUBROUTINE SHIFT (N1,N2,N3,JJ,XBK)                               00020000
   ISN 0003            IMPLICIT REAL*8(A-H,O-Z)                                         00025000
                 C                                                                      00030000
   ISN 0004            DIMENSION XBK(12,12),EXT(9,9),TEMP(9,9)                          00040000
                 C                                                                      00050000
   ISN 0005            DO 1 I =1,9                                                      00060000
   ISN 0006            DO 1 J =1,9                                                      00070000
   ISN 0007          1 EXT(I,J)=0.0                                                     00080000
                 C                                                                      00090000
   ISN 0008            IF ((N2.EQ.JJ).OR.(N3.EQ.JJ)) GO TO 2                            00100000
                 C                                                                      00110000
   ISN 0010            RETURN                                                           00120000
                 C                                                                      00130000
   ISN 0011          2 IF (N2.NE.JJ) GO TO 3                                            00140000
                 C                                                                      00150000
   ISN 0013            DO 4 I =1,3                                                      00160000
   ISN 0014            J=I+3                                                            00170000
   ISN 0015            EXT(I,J)=1.                                                      00180000
   ISN 0016          4 EXT(J,I)=1.                                                      00190000
                 C                                                                      00200000
   ISN 0017            DO 5 I =7,9                                                      00210000
   ISN 0018          5 EXT(I,I)=1.                                                      00220000
                 C                                                                      00230000
   ISN 0019            IB=N1                                                            00240000
   ISN 0020            N1=N2                                                            00250000
   ISN 0021            N2=IB                                                            00260000
   ISN 0022            GO TO 10                                                         00270000
                 C                                                                      00280000
   ISN 0023          3 IF (N3.NE.JJ) GO TO 100                                          00290000
                 C                                                                      00300000
   ISN 0025            DO 6 I =1,3                                                      00310000
   ISN 0026            J=I+6                                                            00320000
   ISN 0027            EXT(I,J)=1.                                                      00330000
   ISN 0028          6 EXT(J,I)=1.                                                      00340000
                 C                                                                      00350000
   ISN 0029            DO 7 I =4,6                                                      00360000
   ISN 0030          7 EXT(I,I)=1.                                                      00370000
   ISN 0031            IB=N1                                                            00380000
   ISN 0032            N1=N3                                                            00390000
   ISN 0033            N3=IB                                                            00400000
                 C                                                                      00410000
   ISN 0034         10 DO 11 I =1,9                                                     00420000
   ISN 0035            DO 11 J =1,9                                                     00430000
                 C                                                                      00440000
   ISN 0036            TEMP(I,J)=0.0                                                    00450000
                 C                                                                      00460000
   ISN 0037            DO 11 K =1,9                                                     00470000
   ISN 0038         11 TEMP(I,J)=TEMP(I,J)+EXT(I,K)*XBK(K,J)                            00480000
```

```
                 C
ISN 0039               DO 12 I =1,9
ISN 0040               DO 12 J =1,9
                 C
ISN 0041               XBK(I,J)=0.0
                 C
ISN 0042               DO 12 K =1,9
ISN 0043            12 XBK(I,J)=XBK(I,J)+TEMP(I,K)*EXT(J,K)
                 C
ISN 0044           100 RETURN
ISN 0045               END
```

```
LEVEL 21.7    JAN 73 )                                OS/360  FORTRAN H

              COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                                 SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                    C     MEMBER NAME STASTRNG                                              00010000
                    C                                                                       00015000
  ISN 0002                SUBROUTINE STRING(X1,Y1,Z1,X2,Y2,Z2,E,A,SMAT,CF1,CF2)             00020000
  ISN 0003                IMPLICIT REAL*8(A-H,O-Z)                                          00025000
  ISN 0004                DOUBLE PRECISION LAM,MU,NU,L                                      00030000
  ISN 0005                DIMENSION SMAT(12,12)                                             00040000
  ISN 0006                SQRT(G)=DSQRT(G)                                                  00041000
                    C                                                                       00050000
  ISN 0007                L=SQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)                          00060000
  ISN 0008                LAM=(X2-X1)/L                                                     00070000
  ISN 0009                MU=(Y2-Y1)/L                                                      00080000
  ISN 0010                NU=(Z2-Z1)/L                                                      00090000
                    C                                                                       00100000
  ISN 0011                SMAT(1,1)=LAM**2                                                  00110000
  ISN 0012                SMAT(2,2)=MU**2                                                   00120000
  ISN 0013                SMAT(3,3)=NU**2                                                   00130000
  ISN 0014                SMAT(4,4)=SMAT(1,1)                                               00140000
  ISN 0015                SMAT(5,5)=SMAT(2,2)                                               00150000
  ISN 0016                SMAT(6,6)=SMAT(3,3)                                               00160000
                    C                                                                       00170000
  ISN 0017                SMAT(1,2)=MU*LAM                                                  00180000
  ISN 0018                SMAT(1,3)=NU*LAM                                                  00190000
  ISN 0019                SMAT(1,4)=-(LAM)**2                                               00200000
  ISN 0020                SMAT(1,5)=-(MU*LAM)                                               00210000
  ISN 0021                SMAT(1,6)=-(NU*LAM)                                               00220000
  ISN 0022                SMAT(2,3)=NU*MU                                                   00230000
  ISN 0023                SMAT(2,4)=-LAM*MU                                                 00240000
  ISN 0024                SMAT(2,5)=-(MU)**2                                                00250000
  ISN 0025                SMAT(2,6)=-NU*MU                                                  00260000
                    C                                                                       00270000
  ISN 0026                SMAT(3,4)=-LAM*NU                                                 00280000
  ISN 0027                SMAT(3,5)=-MU*NU                                                  00290000
  ISN 0028                SMAT(3,6)=-(NU)**2                                                00300000
                    C                                                                       00310000
  ISN 0029                SMAT(4,5)=MU*LAM                                                  00320000
  ISN 0030                SMAT(4,6)=LAM*NU                                                  00330000
                    C                                                                       00340000
  ISN 0031                SMAT(5,6)=MU*NU                                                   00350000
                    C                                                                       00360000
  ISN 0032                DO 100 I=1,6                                                      00370000
  ISN 0033                DO 100 J=I,6                                                      00380000
  ISN 0034            100 SMAT (J,I)=SMAT(I,J)                                              00390000
  ISN 0035                DO 105 I=1,6                                                      00400000
  ISN 0036                DO 105 J=1,6                                                      00410000
  ISN 0037            105 SMAT(I,J)=(A*E)/L*SMAT(I,J)                                       00420000
  ISN 0038                XN1=CF1/L                                                         00430000
  ISN 0039                XN2=CF2/L                                                         00440000
  ISN 0040                SMAT(1,1)=SMAT(1,1)+(1.-LAM*LAM)*XN1                              00450000
  ISN 0041                SMAT(1,2)=SMAT(1,2)-LAM*MU*XN1                                    00460000
  ISN 0042                SMAT(1,3)=SMAT(1,3)-LAM*NU*XN1                                    00470000
```

```
ISN 0043        SMAT(1,4)=SMAT(1,4)-(1.-NU*NU)*XN2                       00480000
ISN 0044        SMAT(1,5)=SMAT(1,5)+LAM*MU*XN2                           00490000
ISN 0045        SMAT(1,6)=SMAT(1,6)+LAM*NU*XN2                           00500000
ISN 0046        SMAT(2,1)=SMAT(2,1)-LAM*MU*XN1                           00510000
ISN 0047        SMAT(2,2)=SMAT(2,2)+(1.-MU*MU)*XN1                       00520000
ISN 0048        SMAT(2,3)=SMAT(2,3)-MU*NU*XN1                            00530000
ISN 0049        SMAT(2,4)=SMAT(2,4)+LAM*MU*XN2                           00540000
ISN 0050        SMAT(2,5)=SMAT(2,5)-(1.-MU*MU)*XN2                       00550000
ISN 0051        SMAT(2,6)=SMAT(2,6)+MU*NU*XN2                            00560000
ISN 0052        SMAT(3,1)=SMAT(3,1)-LAM*NU*XN1                           00570000
ISN 0053        SMAT(3,2)=SMAT(3,2)-MU*NU*XN1                            00580000
ISN 0054        SMAT(3,3)=SMAT(3,3)+(1.-NU*NU)*XN1                       00590000
ISN 0055        SMAT(3,4)=SMAT(3,4)+LAM*NU*XN2                           00600000
ISN 0056        SMAT(3,5)=SMAT(3,5)+MU*NU*XN2                            00610000
ISN 0057        SMAT(3,6)=SMAT(3,6)-(1.-NU*NU)*XN2                       00620000
ISN 0058        DO 106 I=1,3                                             00630000
ISN 0059        DO 106 J=1,6                                             00640000
ISN 0060    106 SMAT(I+3,J)=-SMAT(I,J)                                   00650000
ISN 0061        DO 10 I=1,6                                              00651000
ISN 0062        DO 10 J=1,6                                              00652000
          C                                                              00653500
ISN 0063        IF(DABS(SMAT(I,J)).GT..1) GO TO 10                       00653000
ISN 0065        SMAT(I,J)=0.0                                            00654000
ISN 0066     10 CONTINUE
ISN 0067        RETURN                                                   00660000
ISN 0068        END                                                      00670000
```

```
LEVEL 21.7   JAN 73 )                       OS/360  FORTRAN H

         COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=58,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                 C      MEMBER NAME  S74SKIN                                              00010000
                 C                                                                        00015000
   ISN 0002             SUBROUTINE SKIN (X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,E,G,TS,TX,TY,XMU,TMAT)00020000
   ISN 0003             IMPLICIT REAL*8(A-H,O-Z)                                          00025000
   ISN 0004             DIMENSION TMAT(12,12),D(3,3),OME(9,9),BK(6,6),TEMP(6,9),XX(9),X(6)00030000
   ISN 0005             DOUBLE PRECISION MU                                               00040000
                 C                                                                        00050000
   ISN 0006             SQRT(X)=DSQRT(X)                                                  00051000
   ISN 0007             ABS(X)=DABS(X)                                                    00052000
   ISN 0008             MU=XMU                                                            00060000
   ISN 0009             DD=SQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)                         00070000
   ISN 0010             COSXBX=(X2-X1)/DD                                                 00080000
   ISN 0011             COSYBX=(Y2-Y1)/DD                                                 00090000
   ISN 0012             COSZBX=(Z2-Z1)/DD                                                 00100000
                 C                                                                        00110000
   ISN 0013             I=1                                                               00120000
   ISN 0014             DO 5 J =1,3                                                       00130000
   ISN 0015           5 D(J,I)=1.                                                         00140000
                 C                                                                        00150000
   ISN 0016             D(1,2)=Y1                                                         00160000
   ISN 0017             D(2,2)=Y2                                                         00170000
   ISN 0018             D(3,2)=Y3                                                         00180000
   ISN 0019             D(1,3)=Z1                                                         00190000
   ISN 0020             D(2,3)=Z2                                                         00200000
   ISN 0021             D(3,3)=Z3                                                         00210000
                 C                                                                        00220000
   ISN 0022             CALL DET (D,A1)                                                   00230000
                 C                                                                        00240000
                 C                                                                        00250000
   ISN 0023             D(1,1)=X1                                                         00260000
   ISN 0024             D(2,1)=X2                                                         00270000
   ISN 0025             D(3,1)=X3                                                         00280000
                 C                                                                        00290000
   ISN 0026             I=2                                                               00300000
   ISN 0027             DO 6 J = 1,3                                                      00310000
   ISN 0028           6 D(J,I)= 1.                                                        00320000
                 C                                                                        00330000
   ISN 0029             CALL DET (D,B)                                                    00340000
                 C                                                                        00350000
                 C                                                                        00360000
   ISN 0030             I=3                                                               00370000
   ISN 0031             DO 7 J =1,3                                                       00380000
   ISN 0032           7 D(J,I)=1.                                                         00390000
                 C                                                                        00400000
   ISN 0033             D(1,2)=Y1                                                         00410000
   ISN 0034             D(2,2)=Y2                                                         00420000
   ISN 0035             D(3,2)=Y3                                                         00430000
                 C                                                                        00440000
   ISN 0036             CALL DET (D,C)                                                    00450000
                 C                                                                        00460000
```

```
                C
                C
ISN 0037              G1=SQRT(A1**2+B**2+C**2)
ISN 0038              COSXBZ=A1/G1
ISN 0039              COSYBZ=B/G1
ISN 0040              COSZBZ=C/G1
                C
                C
ISN 0041              COSXBY=COSYBZ*COSZBX-COSYBX*COSZBZ
ISN 0042              COSYBY=COSZBZ*COSXBX-COSZBX*COSXBZ
ISN 0043              COSZBY=COSXBZ*COSYBX-COSXBX*COSYBZ
                C
                C
                C     *** FORM 6X9 MATRIX  UME   ***
                C
                C
ISN 0044              DO  8 I =1,6
ISN 0045              DO  9 J =1,9
ISN 0046            9 UME(I,J)=0.0
ISN 0047            8 CONTINUE
                C
ISN 0048              UME(1,1)=COSXBX
ISN 0049              UME(1,2)=COSYBX
ISN 0050              UME(1,3)=COSZBX
                C
ISN 0051              UME(2,1)=COSXBY
ISN 0052              UME(2,2)=COSYBY
ISN 0053              UME(2,3)=COSZBY
                C
ISN 0054              UME(3,4)=COSXBX
ISN 0055              UME(3,5)=COSYBX
ISN 0056              UME(3,6)=COSZBX
ISN 0057              UME(4,4)=COSXBY
ISN 0058              UME(4,5)=COSYBY
ISN 0059              UME(4,6)=COSZBY
                C
ISN 0060              UME(5,7)=COSXBX
ISN 0061              UME(5,8)=COSYBX
ISN 0062              UME(5,9)=COSZBX
                C
ISN 0063              UME(6,7)=COSXBY
ISN 0064              UME(6,8)=COSYBY
ISN 0065              UME(6,9)=COSZBY
                C
                C     *** FORM 6X6 MATRIX BK ***
                C
ISN 0066              XX(1)=X1
ISN 0067              XX(2)=Y1
ISN 0068              XX(3)=Z1
ISN 0069              XX(4)=X2
ISN 0070              XX(5)=Y2
ISN 0071              XX(6)=Z2
```

```
ISN 0072          XX(7)=X3                                        00990000
ISN 0073          XX(8)=Y3                                        01000000
ISN 0074          XX(9)=Z3                                        01010000
          C                                                       01020000
ISN 0075          DO 70 I = 1,6                                   01030000
ISN 0076          X(I)=0.0                                        01040000
ISN 0077          DO 70 J = 1,9                                   01050000
ISN 0078       70 X(I)=X(I)+OME(I,J) *XX(J)                       01060000
ISN 0079          X12=X(1)-X(3)                                   01070000
ISN 0080          X13=X(1)-X(5)                                   01080000
ISN 0081          X21=-X12                                        01090000
ISN 0082          X23=X(3)-X(5)                                   01100000
ISN 0083          X31=-X13                                        01110000
ISN 0084          X32=X(5)-X(3)                                   01120000
ISN 0085          Y12=X(2)-X(4)                                   01130000
ISN 0086          Y13=X(2)-X(6)                                   01140000
ISN 0087          Y21=-Y12                                        01150000
ISN 0088          Y31=-Y13                                        01160000
ISN 0089          Y23=X(4)-X(6)                                   01170000
ISN 0090          Y32=-Y23                                        01180000
          C                                                       01190000
          C                                                       01200000
ISN 0091          A=.5*(X21*Y31-Y21*X31)                          01210000
ISN 0092          GO=TS*G/(4.*A)                                  01220000
          C                                                       01230000
ISN 0093          GX=MU                                           01240000
ISN 0094          GY=0.                                           01241000
ISN 0095          IF(TY.EQ.0.) GO TO 105                          01242000
ISN 0097          GY=MU*TX/TY                                     01250000
          C                                                       01260000
ISN 0098      105 IF(TX.LT.TY) GO TO 50                           01270000
          C                                                       01280000
ISN 0100          GX=0.                                           01281000
ISN 0101          IF(TX.EQ.0.) GO TO 200                          01282000
ISN 0103          GX=MU*TY/TX                                     01290000
ISN 0104      200 GY=MU                                           01300000
          C                                                       01310000
ISN 0105       50 CONTINUE                                        01320000
          C                                                       01330000
ISN 0106          EY=TY*E/(4.*A*(1.-GX*GY))                       01340000
ISN 0107          EX=TX*E/(4.*A*(1.-GX*GY))                       01350000
          C                                                       01360000
ISN 0108          BK(1,1)=EX*Y23**2+GO*X32**2                     01370000
ISN 0109          BK(2,1)=Y23*X32*(EY*GY+GO)                      01380000
ISN 0110          BK(3,1)=EX*Y31*Y23+GO*X13*X32                   01390000
ISN 0111          BK(4,1)=EY*GY*X13*Y23+GO*Y31*X32                01400000
ISN 0112          BK(5,1)=EX*Y12*Y23+GO*X21*X32                   01410000
ISN 0113          BK(6,1)=EY*GY*X21*Y23+GO*Y12*X32                01420000
          C                                                       01430000
ISN 0114          BK(2,2)=EY*X32**2+GO*Y23**2                     01440000
ISN 0115          BK(3,2)=EX*GX*Y31*X32+GO*X13*Y23                01450000
ISN 0116          BK(4,2)=EY*X13*X32+GO*Y31*Y23                   01460000
```

```
ISN 0117          BK(5,2)=EX*GX*Y12*X32+GO*X21*Y23
ISN 0118          BK(6,2)=EY*X21*X32+GO*Y12*Y23
            C
ISN 0119          BK(3,3)=EX*Y31**2+GO*X13**2
ISN 0120          BK(4,3)=EY*GY*X13*Y31+GO*Y31*X13
ISN 0121          BK(5,3)=EX*Y12*Y31+GO*X21*X13
ISN 0122          BK(6,3)=EY*GY*X21*Y31+GO*Y12*X13
            C
ISN 0123          BK(4,4)=EY*X13**2+GO*Y31**2
ISN 0124          BK(5,4)=EX*GX*X13*Y12+GO*X21*Y31
ISN 0125          BK(6,4)=EY*X21*X13+GO*Y12*Y31
            C
ISN 0126          BK(5,5)=EX*Y12**2+GO*X21**2
ISN 0127          BK(6,5)=X21*Y12*(EY*GY+GO)
            C
ISN 0128          BK(6,6)=EY*X21**2+GO*Y12**2
            C
ISN 0129          DO 100 I =1,6
ISN 0130          DO 100 J =I,6
ISN 0131      100 BK(I,J)=BK(J,I)
ISN 0132          DO 10 I =1,6
ISN 0133          DO 10 J =1,9
ISN 0134          TEMP(I,J)=0.0
ISN 0135          DO 10 K =1,6
            C
ISN 0136       10 TEMP(I,J)=TEMP(I,J)+BK(I,K)*OME(K,J)
            C
ISN 0137          DO 11 I =1,9
ISN 0138          DO 11 J =1,9
ISN 0139          TMAT(I,J)=0.0
ISN 0140          DO  5 K =1,6
            C
ISN 0141        5 TMAT(I,J)=TMAT(I,J)+OME(K,I)*TEMP(K,J)
ISN 0142          IF (ABS(TMAT(I,J)).GT..1) GO TO 11
ISN 0144          TMAT(I,J)=0.0
ISN 0145       11 CONTINUE
ISN 0146          RETURN
ISN 0147          END
```

```
LEVEL 21    ( JAN 73 )                                   OS/360  FORTRAN H

              COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                                 SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                        C        MEMBER NAME S74DET                                              00010000
                        C                                                                        00015000
  ISN 0002                       SUBROUTINE DET (D,ANS)                                          00020000
  ISN 0003                       IMPLICIT REAL*8(A-H,O-Z)                                        00025000
                        C                                                                        00030000
  ISN 0004                       DIMENSION D(3,3)                                                00040000
                        C                                                                        00050000
  ISN 0005                       ANS=D(1,1)*D(2,2)*D(3,3)+D(1,2)*D(2,3)*D(3,1)+D(1,3)*D(2,1)*D(3,2)00060000
  ISN 0006                       ANS=ANS-(D(3,1)*D(2,2)*D(1,3)+D(3,2)*D(2,3)*D(1,1)+D(3,3)*D(2,1)* 00070000
                            1              D(1,2))                                               00080000
                        C                                                                        00090000
  ISN 0007                       RETURN                                                          00100000
  ISN 0008                       END                                                             00110000
```

```
          COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                   C      MEMBER NAME S74RDLK2                                        00010000
                   C                                                                  00015000
  ISN 0002                SUBROUTINE REDLK2                                           00020000
  ISN 0003                IMPLICIT REAL*8(A-H,O-Z)                                    00021000
  ISN 0004                COMMON/LINKA1/NTOTAL,NORET,NTAPE                            00030000
  ISN 0005                DIMENSION NMAP(3000),FIX(3000),RO(3000)                     00040000
  ISN 0006            100 FORMAT( //10X,'OLMEC=',I3,5X,'TIME=',E13.6)                 00042000
  ISN 0007                NN=NTOTAL                                                   00050000
                   C                                                                  00060000
  ISN 0008                REWIND 5                                                    00070000
  ISN 0009                READ  (5) (NMAP(JTK),JTK=1,NTOTAL)                          00080000
  ISN 0010                NTAPEA=1                                                    00090000
  ISN 0011                NTAPEB=3                                                    00100000
                   C                                                                  00110000
  ISN 0012                REWIND  NTAPEA                                              00120000
  ISN 0013                REWIND  NTAPEB                                              00130000
                   C                                                                  00140000
                   C                                                                  00150000
  ISN 0014                IF (NTOTAL.NE.NORET) GO TO 555                              00160000
  ISN 0016                NTAPE=NTAPEA                                                00170000
                   C                                                                  00180000
                   C                                                                  00190000
  ISN 0017                RETURN                                                      00200000
                   C                                                                  00210000
  ISN 0018            555 CONTINUE                                                    00220000
                   C                                                                  00230000
  ISN 0019                CALL LOADER (SKP,2)                                         00260000
                   C                                                                  00300000
  ISN 0020                KO=0                                                        00301000
  ISN 0021            500 N=NN-1                                                      00310000
                   C                                                                  00320000
                   C  ----------------------------FIX IS THE ROW TO BE REDUCED OUT    00330000
                   C                                                                  00340000
  ISN 0022                KO= KO+1                                                    00341000
  ISN 0023                IF(KO.EQ.1 ) WRITE(6,887) NN                                00350000
  ISN 0025            887 FORMAT (9X,4HROW=I5/)                                       00360000
                   C                                                                  00370000
  ISN 0026                ISEND=NMAP(NN)                                              00380000
  ISN 0027                READ (NTAPEA) (FIX(I),I=1,ISEND )                           00390000
  ISN 0028                ISST=1                                                      00400000
  ISN 0029                IBEND=NN                                                    00410000
                   C                                                                  00420000
  ISN 0030                CALL ZIN (FIX,ISST,ISEND,IBEND)                             00430000
                   C                                                                  00440000
                   C                                                                  00640000
                   C                                                                  00650000
  ISN 0031                IF(FIX(NN).NE.0.) GO TO 61                                  00660000
                   C                                                                  00670000
  ISN 0033                IF(NN.NE.NORET+1) GO TO 60                                  00671000
  ISN 0035                NNN=N                                                       00672000
```

```
ISN 0036            DO 40 I=1,N                                          00673000
ISN 0037            ISEND=NMAP(NNN)                                      00675000
ISN 0038            READ(NTAPEA) (RO(K),K=1,ISEND)                       00677000
ISN 0039            IBEND=NNN                                            00679000
ISN 0040            CALL ZIN(RO,1,ISEND,IBEND)                           00680000
ISN 0041            WRITE(NTAPEB) (RO(J),J=1,NNN)                        00681000
ISN 0042         40 NNN=NNN-1                                            00683000
ISN 0043            GO TO 50                                             00686000
ISN 0044         60 NN=NN-1                                              00689000
ISN 0045            GO TO 500                                            00690000
              C                                                          00700000
              C                                                          00730000
ISN 0046         61 CONTINUE                                             00740000
              C                                                          00750000
ISN 0047            CALL REDUCE(FIX,N,NN,NTAPEA,NTAPEB,RO,NMAP,NORET)    00760000
              C                                                          00770000
              C                                                          00780000
              C                                                          00830000
ISN 0048         50 NN=NN-1                                              00840000
ISN 0049            REWIND  NTAPEA                                       00850000
ISN 0050            REWIND  NTAPEB                                       00860000
              C                                                          00870000
ISN 0051            IF(NN.EQ.NORET) GO TO 200                            00880000
              C                                                          00890000
              C                         SWAP SWINGING METHOD             00900000
              C                                                          00910000
ISN 0053            NDUM=NTAPEB                                          00920000
ISN 0054            NTAPEB=NTAPEA                                        00930000
ISN 0055            NTAPEA=NDUM                                          00940000
ISN 0056            IF(KO.LE.9) KO=0                                     00941000
ISN 0058            GO TO 500                                            00950000
ISN 0059        200 CONTINUE                                             00960000
ISN 0060            NTAPE=NTAPEB                                         00970000
ISN 0061            RETURN                                               00980000
ISN 0062            END                                                  00990000
```

```
LEVEL 21.8   ( JAN 73 )                                   OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                  C     MEMBER NAME 374REDUC                                              00010000
                  C                                                                       00020000
 ISN 0002               SUBROUTINE REDUCE(FIX,N,NN,NTAPEA,NTAPEB,RO,NMAP,NORET)           00030000
 ISN 0003               IMPLICIT REAL*8(A-H,O-Z)                                          00040000
 ISN 0004               DIMENSION RO(1),FIX(1),NMAP(1)                                    00050000
 ISN 0005               A=FIX(NN)                                                         00060000
 ISN 0006               NNN=N                                                             00070000
 ISN 0007               NRT=NORET+1                                                       00080000
 ISN 0008               DO 9 I=1,N                                                        00090000
 ISN 0009               ISEND=NMAP(NNN)                                                   00100000
 ISN 0010               READ(NTAPEA) (RO(K),K=1,ISEND)                                    00110000
 ISN 0011               IF(FIX(NNN).EQ.0..AND.NN.NE.NRT) GO TO 78                         00120000
 ISN 0013               IBEND=NNN                                                         00130000
 ISN 0014               CALL ZIN(RO,1,ISEND,IBEND)                                        00140000
 ISN 0015               II=1                                                              00150000
 ISN 0016               B=FIX(NNN)                                                        00160000
 ISN 0017               AA=B/A                                                            00170000
 ISN 0018               DO 11 J=1,NNN                                                     00180000
 ISN 0019               RO(II)=RO(II)-FIX(J)*AA                                           00190000
 ISN 0020            11 II=II+1                                                           00200000
 ISN 0021               IBUB=NNN                                                          00210000
 ISN 0022               IF(NN.EQ.NRT) GO TO 88                                            00220000
 ISN 0024               CALL ZOUT2(RO,1,IBDB,NMAP)                                        00230000
 ISN 0025            78 IBOB=NMAP(NNN)                                                    00240000
 ISN 0026            88 WRITE(NTAPEB) (RO(L),L=1,IBUB)                                    00250000
 ISN 0027             9 NNN=NNN-1                                                         00260000
 ISN 0028               RETURN                                                            00270000
 ISN 0029               END                                                               00280000
```

```
LEVEL 21.7   JAN 73 )                              OS/360  FORTRAN H

         COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                 C      MEMBER NAME S74ZIN                                              00010000
                 C                                                                      00015000
  ISN 0002              SUBROUTINE ZIN (RO,ISST,ISEND,IBEND)                            00020000
  ISN 0003              IMPLICIT REAL*8(A-H,O-Z)                                        00025000
  ISN 0004              DIMENSION RO(1)                                                 00030000
  ISN 0005              ABS(X)=DABS(X)                                                  00031000
  ISN 0006           50 IF (ABS(RO(ISEND)).GT.1.D-2) GO TO 40                           00040000
                 C                                                                      00050000
  ISN 0008              NRS=RO(ISEND)*1.D6+.1                                           00060000
                 C                                                                      00070000
  ISN 0009              DO 20 I =1,NRS                                                  00080000
                 C                                                                      00090000
  ISN 0010              RO(IBEND)=0.0                                                   00100000
  ISN 0011              IF (I.EQ.NRS) GO TO 30                                          00110000
  ISN 0013           20 IBEND=IBEND-1                                                   00120000
                 C                                                                      00130000
                 C                                                                      00140000
  ISN 0014           40 RO(IBEND)=RO(ISEND)                                             00150000
                 C                                                                      00160000
  ISN 0015           30 IF (ISEND.EQ.ISST) RETURN                                       00170000
                 C                                                                      00180000
                 C                                                                      00190000
  ISN 0017              ISEND=ISEND-1                                                   00200000
  ISN 0018              IBEND=IBEND-1                                                   00210000
  ISN 0019              GO TO 50                                                        00220000
                 C                                                                      00230000
  ISN 0020              END                                                             00240000
```

```
LEVEL 21.7   JAN 73 )                                   OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                  C     SOURCE MEMBER NAME     D82ZOUT2                          00010000
                  C     SOURCE MEMBER        S74ZOUT2                            00020000
  ISN 0002              SUBROUTINE ZOUT2(RO,IA,NN,NMAP)                          00030000
  ISN 0003              IMPLICIT REAL*8(A-H,O-Z)                                 00040000
  ISN 0004              DIMENSION RO(1),NMAP(1)                                  00050000
  ISN 0005              ABS(X)=DABS(X)                                           00060000
  ISN 0006              FLOAT(L)=DFLOAT(L)                                       00070000
  ISN 0007              M=IA                                                     00080000
  ISN 0008              MM=0                                                     00090000
  ISN 0009              DO 10 I = IA,NN                                          00100000
  ISN 0010              IF(ABS(RO(I)).GT.0.IO) GO TO 5                           00110000
  ISN 0012              MM = MM+1                                                00120000
  ISN 0013              GO TO 10                                                 00130000
  ISN 0014            5 IF(MM.EQ.0) GO TO 7                                      00140000
  ISN 0016              RO(M) = FLOAT(MM)*1.D-6                                  00150000
  ISN 0017              M = M + 1                                                00160000
  ISN 0018            7 RO(M) = RO(I)                                            00170000
  ISN 0019              M = M+1                                                  00180000
  ISN 0020              MM = 0                                                   00190000
  ISN 0021           10 CONTINUE                                                 00200000
  ISN 0022              IF(MM.EQ.0)M=M-1                                         00210000
  ISN 0024              IF(MM.NE.0) RO(M)=FLOAT(MM)*1.D-6                        00220000
  ISN 0026              NMAP(NN)=M-IA+1                                          00230000
  ISN 0027              RETURN                                                   00240000
  ISN 0028              END                                                      00250000
```

```
.EVEL 21.*   JAN 73 )                          OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
               C     MEMBER NAME S74A6
               C                                                                        00010000
               C                 EIGENVALUES & EIGENVECTORS BY USING ROUTINES           00030000
               C                 EIGEN & GELS FROM THE SSP.                             00040000
               C                                                                        00070000
               C                       MEMBER=NAME U82A6                                00080000
               C                 ADDED TO SM1 ON 10TH, MARCH, 1969                      00090000
               C                                                                        00100000
               C                                                                        00110000
  ISN 0002           SUBROUTINE ABLNK3                                                  00120000
               C                                                                        00130000
               C                                                                        00140000
  ISN 0003           DOUBLE PRECISION DSROUT,IMASS,SROUT,SCALF,ROOIR,RORD,TESE,SUM2,    00145000
                    1 SQRT,TIME,DARK,TOP,SUM,SKP,RHS,HGB,EPS,DAR,AUX,ABS,XM,XK,T1,R,    00147000
                    2 ROWN,OR,XI,SUMA,ROOTS,DROOTS,PHK,PHIK,DAR1,X                      00148000
  ISN 0004           COMMON /LINKA/ LAP(250),                                           00150000
                    1         XNONOD,XNOSAT,XNORET,XNORED,XNOPOP,AXNOD,EE(3),XNU(3),    00160000
                    2 XNMASV,XNEIG,XNMASG,SCALE,EVSAVE,XKSAVE,XNSPEC,                   00170000
                    3 AS,FSM,RSM,UM,UR,RONE,OINV,DUM(5),RONRM                           00180000
               C                                                                        00190000
  ISN 0005           COMMON/LINKA1/NTOTAL,NORET,NTAPE                                   00200000
               C                                                                        00210000
  ISN 0006           DIMENSION AUX(250),DARK(250,3),HGB(30),MAP(250),PACK(6),           00220000
                    1           RHS(250),ROUTR(250),IMASS(6),WORD(2),XM(250),           00230000
                    2           XK(23005),PHIK(3)                                       00240000
               C                                                                        00250000
               C                                                                        00270000
  ISN 0007           EQUIVALENCE (LAP(1),MAP(1))                                        00280000
               C                                                                        00290000
               C                                                                        00300000
  ISN 0008       901 FORMAT(1H0,20X,'ARO STRAIN ENERGY VIBRATION OPTIMIZATION'/1H ,54X,00310000
                    *'PROGRAM S-74'/1H ,25X,'CONTRACT NO. DAHC04-71-C-0048'/1H ,30X,'SC00320000
                    *IARRA/RODRIGUEZ'/1H0,25X,'BOEING COMPANY VERTOL DIVISION.'//)      00321000
  ISN 0009       902 FORMAT( 1H0, 21X ' RETAINED STIFFNESS MATRIX' /)                   00330000
  ISN 0010       903 FORMAT( 1H0, 27X 'FLEXIBILITY MATRIX' /)                           00340000
  ISN 0011       904 FORMAT ( 1X 80(1H-))                                               00350000
  ISN 0012       906 FORMAT (I4,1X,6E10.0,2A6)                                          00370000
  ISN 0013       907 FORMAT (2X,I3,1X,6E11.4,1X,2A6)                                    00380000
  ISN 0014       908 FORMAT ( 1X 80(1H-) // 30X 13H EIGENVALUES )                       00390000
  ISN 0015       910 FORMAT (20X,11HEIGENVALUE=F15.6,13H,ASSUMED ZERO)                  00410000
  ISN 0016       911 FORMAT( / 30X 'EIGENVALUE' // 25X E20.7 // 50X 'EIGENVECTORS' )    00420000
  ISN 0017       912 FORMAT( / 40X 'EIGENVALUES' // 23X 2E20.7 // 40X  'EIGENVECTORS')  00430000
  ISN 0018       913 FORMAT( / 50X 'EIGENVALUES' // 23X 3E20.7 // 50X 'EIGENVECTORS' )  00440000
  ISN 0019      914  FORMAT( 7X,'ROW',7X,'NODE'/)                                       00450000
  ISN 0020       915 FORMAT( 2I9, 5X 3E20.7)                                            00460000
  ISN 0021       916 FORMAT (18X,21H NATURAL FREQUENCIES )                              00470000
  ISN 0022       917 FORMAT (18X,17H NEG. EIGENVALUE E15.6)                             00480000
  ISN 0023       918 FORMAT (1X,E17.8,2X,7HRAD/SEC,3X,13HEIGENVALUE = E16.8,            00490000
                    1              3X 5HCPS =E13.6 )                                    00500000
  ISN 0024       919 FORMAT ( 1H1,10X, 'SUBROUTINE GELS WARNING = ',  /// 10X           00510000
```

```
                  1          'ERROR PARAMETER IER = -1 DOES NOT NECESSAR',            00520000
                  2          'ILY MEAN', / 10X, ' THAT THE MATRIX IS SING',           00530000
                  3          'ULAR,AS ONLY MAIN DIAGONAL', / 10X, 'ELEMENTS',         00540000
                  4          ' ARE USED AS PIVOT ELEMENTS.' )                         00550000
ISN 0025      920 FORMAT ( 1H1,10X, 'SUBROUTINE GELS WARNING = ',  /// 10X            00560000
                  1          'A POSSIBLE LOSS OF SIGNIFICANCE IS INDICATED,'/         00570000
                  2          10X,'WITH IER =',I3,', IN THE CASE OF A WELL-',          00580000
                  3          'SCALED MATRIX',/ 10X, 'AND APPROPRIATE TOLERANCE',      00590000
                  4          ' EPS,IT MAY BE INTERPRETED ' / 10X,                     00600000
                  5          'THAT THE MATRIX HAS RANK',I3,'.' )                      00610000
ISN 0026      924 FORMAT ( 14X,I4,3X,I4,3(3X,E15.6) )                                 00670000
ISN 0027      925 FORMAT( 1H1, //// 20X,'EFFECTIVE MASS       =',E15.6 )             00680000
ISN 0028      926 FORMAT  ( //// 20X, 'TOTAL EFFECTIVE', /                            00690000
                1 20X,'STRAIN ENERGY       =',E15.6)                                  00700000
ISN 0029      927 FORMAT  ( //// 20X, 'EFFECTIVE STRAIN', /                           00710000
                1 20X,'ENERGY RESIDUAL     =',E15.6)                                  00720000
ISN 0030      928 FORMAT(/// 10X,'THIS APPLIES TO EIGENVALUES NO.',I3 )               00730000
ISN 0031      929 FORMAT  ( 1H1,/// 16X,'THE KR MATRIX WAS SINGULAR FOR THIS CASE' )00740000
ISN 0032      930 FORMAT(1H1,///,15X,'N TOOK ON THE ILLEGAL VALUE OF ', I7 )          00750000
ISN 0033      932 FORMAT  ( 1H1, 31X, 'OFF-DIAGONAL  STRAIN', / 34X,                  00770000
                  1          'CONTRIBUTIONS' /// 16X, 'NODE', 6X,'NODE',              00780000
                  2          5X, 'ROW', 5X, 'COLUMN', 7X, 'ELEMENT' / )               00790000
ISN 0034      933 FORMAT  ( 16X, I5, 4X, I5, 4X, I5, 4X, I5, 4X, E15.6 )              00800000
ISN 0035      935 FORMAT( // 30X,'EIGENVECTORS NORMALIZED WITH RESPECT TO ROW ',I3) 00805000
ISN 0036      936 FORMAT( // 25X,'EIGENVECTORS NORMALIZED WITH RESPECT TO THE MAXIMU00806000
                1M ELEMENT')                                                          00807000
ISN 0037      937 FORMAT(////,20X,'EFFECTIVE MASS (',I1,') =',E13.6/                 00808000
                1/ 20X,'EFFECTIVE STIFFNESS (',I1,') =',E13.6)                        00808500
            C                                                                         00810000
            C                                                                         00820000
            C        NOMASV= NUMBER OF MASS VARIATIONS FOR STUDY                      00830000
            C        XNEIG = NUMBER OF EIGENVECTOR WANTED                             00840000
            C        NOMASG= NUMBER OF MASS GROUPS                                    00850000
            C        SCALE = SCALE FACTOR                                             00860000
            C                                                                         00870000
ISN 0038             ABS(X)=DABS(X)                                                   00875000
ISN 0039             SQRT(X)=DSQRT(X)                                                 00876000
ISN 0040             SCALE = 1.                                                       00880000
ISN 0041             SCALF = SCALE                                                    00885000
ISN 0042             NOMASV=XNMASV                                                    00890000
ISN 0043             NRMRU=NONKM+.1                                                   00895000
ISN 0044             NN=XNEIG                                                         00900000
ISN 0045             NOMASG=XNMASG                                                    00910000
ISN 0046             IF (HUMASV,EW,0) NOMASV=1                                        00920000
ISN 0048             N=NORET                                                          00930000
ISN 0049             NSU  =  N* (N+1)/2                                               00940000
ISN 0050             MRI  =  N* (N-1)/2                                               00950000
ISN 0051             REWIND 1                                                         00960000
ISN 0052             REWIND 3                                                         00970000
ISN 0053             REWIND 12                                                        00980000
ISN 0054             DO 10 I=1,N                                                      00990000
ISN 0055             J  = N-I +1                                                      01000000
```

```
ISN 0056              J2 = J* (J+1)/2                                01010000
ISN 0057              J1 = J2-J +1                                   01020000
ISN 0058              READ (NTAPE) (XK(J),J=J1,J2)                   01030000
ISN 0059              WRITE(12)(XK(J),J=J1,J2)                       01031000
ISN 0060           10 CONTINUE                                       01040000
ISN 0061              REWIND NTAPE                                   01050000
ISN 0062              MRIX = 1                                       01060000
ISN 0063              IF(KSM.EQ.1.) GO TO 30                         01070000
ISN 0065           20 CALL LOADER(FLIP,2)                            01080000
ISN 0066              WRITE (6,901)                                  01090000
ISN 0067              IF(MRIX.EQ.1) WRITE(6,902)                     01100000
ISN 0069              IF(MRIX.EQ.2) WRITE(6,903)                     01110000
ISN 0071              CALL  MXOUT (MRIX,XK,N,N,1,50,80,1,6)          01120000
ISN 0072           30 IF (MRIX.EQ.2)  GO TO 60                       01130000
ISN 0074              IF (NUMASG.EQ.0) GO TO 440                     01140000
ISN 0076              REWIND NTAPE                                   01150000
ISN 0077              WRITE (3) (XK(I),I=1,NSU)                      01160000
ISN 0078              REWIND 3                                       01170000
ISN 0079              IF(DINV.NE.1.) GO TO 60                        01180000
ISN 0081              READ  (3) (XK(I),I=1,NSU)                      01190000
ISN 0082              REWIND 3                                       01200000
ISN 0083              MRIX=2                                         01210000
ISN 0084              CALL  INVA(XK,N,IER)                           01220000
ISN 0085              IF (IER-1)  40,20,50                           01230000
ISN 0086           40 WRITE (6,929)                                  01240000
ISN 0087              GO TO 20                                       01250000
ISN 0088           50 WRITE (6,930)  N                               01260000
ISN 0089              GO TO 20                                       01270000
            C                                                        01280000
ISN 0090           60 READ  (3) (XK(I),I=1,NSU)                      01290000
ISN 0091              REWIND 3                                       01300000
ISN 0092              IB  = 1                                        01350000
ISN 0093              KOUNT=1                                        01360000
ISN 0094           70 READ (5,906) NODE,TMASS,WORD                   01370000
            C                                                        01380000
ISN 0095              DO 72 J=1,250                                  01392000
ISN 0096              IF(MAP(J).NE.NODE) GO TO 72                    01393000
ISN 0098              IB=J                                           01394000
ISN 0099              GO TO 75                                       01395000
ISN 0100           72 CONTINUE                                       01397000
ISN 0101           75 DO 80 I=1,6                                    01400000
ISN 0102              IF (TMASS(I).EQ.0)  GO TO 90                   01410000
ISN 0104              XM(IB) = TMASS(I)                              01420000
ISN 0105           80 IB = IB +1                                     01430000
ISN 0106           90 IF (KOUNT.GE.NUMASG) GO TO 100                 01440000
ISN 0108              KOUNT=KOUNT+1                                  01450000
ISN 0109              GO TO 70                                       01460000
            C                                                        01470000
ISN 0110          100 CALL SJB(XK,XM,SCALF,N)                        01480000
ISN 0111              REWIND 1                                       01490000
ISN 0112              WRITE (1) (XK(I),I=1,NSU)                      01500000
ISN 0113              CALL LOADER (SKP,2)                            01590000
```

```
ISN 0114          WRITE (6,901)                                                     01600000
ISN 0115          WRITE (6,906)                                                     01610000
ISN 0116          CALL EIGEN(XX,R,N,1)                                              01650000
ISN 0117          DO 120 I=1,N                                                      01660000
ISN 0118          IS = I*(I+1)/2                                                    01670000
ISN 0119          ROOTR(I) = XK(IS)                                                 01680000
ISN 0120      120 CONTINUE                                                          01690000
            C                                                                       01700000
            C     *****   PUT EIGENVALUES IN ASCENDING ORDER   ******               01710000
            C                                                                       01720000
ISN 0121          DO 130  IR=1,N                                                    01730000
ISN 0122      130 ROOTR(IR)=ROOTR(IR)* SCALF                                        01740000
            C                                                                       01750000
ISN 0123          DO 150  I=1,N                                                     01760000
ISN 0124          DO 140  J=I,N                                                     01770000
ISN 0125          IF (ROOTR(J).GT.ROOTR(I)) GO TO 140                               01780000
ISN 0127          T1=ROOTR(I)                                                       01790000
ISN 0128          ROOTR(I)=ROOTR(J)                                                 01800000
ISN 0129          ROOTR(J)=T1                                                       01810000
ISN 0130      140 CONTINUE                                                          01820000
ISN 0131      150 CONTINUE                                                          01830000
ISN 0132          DO 500 I=1,N                                                      01840000
ISN 0133          IF(ROOTR(I).GT.0.0) GO TO 510                                     01843000
ISN 0135          WRITE(6,917) ROOTR(I)                                             01846000
ISN 0136          GO TO 500                                                         01848000
ISN 0137      510 ROOTS = SQRT(ROOTR(I))                                            01850000
ISN 0138          DROOTS = ROOTS / 6.28316                                          01855000
ISN 0139          WRITE(6,918) ROOTS,ROOTR(I),DROOTS                                01860000
ISN 0140      500 CONTINUE                                                          01863000
            C                                                                       01870000
            C********************************************************************01880000
            C                       START OF FINDING EIGENVECTOR                    01890000
            C********************************************************************01900000
            C                                                                       01910000
ISN 0141          L=0                                                               01920000
ISN 0142          WRITE (6,900)                                                     01930000
ISN 0143          IF (NN.LE.0)  GO TO 400                                           01940000
ISN 0145      170 L=L+1                                                             01950000
            C                                                                       01960000
ISN 0146          IF(ROOTR(L).GT.WONE) GO TO 180                                    01970000
ISN 0148          WRITE (6,910) ROOTR(L)                                            01980000
ISN 0149          GO TO 170                                                         01990000
ISN 0150      180 REWIND 1                                                          02000000
ISN 0151          READ  (1) (XK(I),I=1,NSU)                                         02010000
ISN 0152          REWIND 1                                                          02020000
            C                                                                       02030000
            C                                                                       02040000
ISN 0153          K=0                                                               02050000
ISN 0154          NKN=3                                                             02060000
ISN 0155          I2= N-1                                                           02070000
ISN 0156      190 CONTINUE                                                          02080000
ISN 0157          IF ((NN-K).LE.2) NKN=NN-K                                         02090000
```

```
ISN 0159          DO 330 KN=1,NKN                                    02100000
ISN 0160          READ (1) (XK(I),I=1,NSU)                           02110000
ISN 0161          REWIND 1                                           02120000
ISN 0162          DO 200 I=1,I2                                      02130000
ISN 0163          IS = I+ MRT                                        02140000
ISN 0164          RHS(I) = -XK(IS)                                   02150000
ISN 0165      200 CONTINUE                                           02160000
ISN 0166          K=K+1                                              02170000
ISN 0167          DO 210 I = 1,I2                                    02180000
ISN 0168          IS  =  I*(I+1)/2                                   02190000
ISN 0169      210 XK(IS) = XK(IS)-ROOTR(L)                           02200000
ISN 0170          HGB(K) = ROOTR(L)                                  02210000
        C                                                            02220000
        C                                                            02230000
ISN 0171          EPS = 1.0E-07                                      02240000
ISN 0172          CALL  GELS (RHS,XK,I2,1,EPS,IER,AUX)               02250000
ISN 0173          IF (IER)  220,240,230                              02260000
ISN 0174      220 WRITE (6,919)                                      02290000
ISN 0175          WRITE (6,928)  L                                   02300000
ISN 0176          GO TO 240                                          02310000
ISN 0177      230 WRITE (6,920) IER,IER                              02320000
ISN 0178          WRITE (6,928)  L                                   02330000
ISN 0179      240 RHS(N) = 1.0                                       02340000
ISN 0180          SUM  =  0.0                                        02350000
ISN 0181          DO 250 I=1,N                                       02360000
ISN 0182          RHS(I) = RHS(I)*XM(I)                              02370000
ISN 0183      250 SUM  = SUM  +  RHS(I)**2                           02380000
ISN 0184          SUM=SQRT(SUM)                                      02390000
        C                                                            02400000
ISN 0185          DO 260 I =1,N                                      02410000
ISN 0186      260 RHS(I) = RHS(I)/SUM                                02420000
ISN 0187          DO 270  I=1,N                                      02430000
ISN 0188      270 DARK(I,KN) = RHS(I)                                02440000
        C                                                            02450000
ISN 0189          IF(EVSAVE.NE.1.) GO TO 280                         02460000
ISN 0191          WRITE (3) (RHS(I),I=1,N)                           02470000
        C                                                            02480000
ISN 0192      280 CONTINUE                                           02490000
ISN 0193          SUM  = 0                                           02510000
ISN 0194          SUM2 = 0                                           02520000
ISN 0195          TUP = -1.0E+10                                     02530000
ISN 0196          READ (1)  (XK(I),I=1,NSU)                          02540000
ISN 0197          REWIND 1                                           02550000
ISN 0198          DO 290 I=1,N                                       02560000
ISN 0199          J = I*(I+1)/2                                      02570000
ISN 0200          XI = RHS(I)                                        02580000
ISN 0201          DAR1 = XI / (XM(I)*XM(I))                          02585000
ISN 0202          DAR = XI * DAR1                                    02590000
ISN 0203          TESE=XK(J)*XI*XI                                   02600000
ISN 0204          IF (TUP.LT.TESE)  TUP = TESE                       02610000
ISN 0206          SUM  = SUM + DAR                                   02620000
ISN 0207          SUM2 = SUM2 + TESE                                 02630000
```

```
ISN 0208    290 CONTINUE                                          02690000
ISN 0209    600 WRITE(6,925) SUM                                  02700000
ISN 0210        TESE = SUM*ROOTR(L)                               02710000
ISN 0211        WRITE (6,926)  TESE                               02720000
ISN 0212        PHIK(KN) = TESE                                   02725000
ISN 0213        SUM2 = TESE - SUM2                                02730000
ISN 0214        WRITE (6,927)  SUM2                               02740000
ISN 0215    325 IN=KN                                             02920000
ISN 0216    330 L=L+1                                             02930000
C                                                                 02940000
ISN 0217        NL=L-NKN                                          02950000
ISN 0218        K1=L-1                                            02960000
ISN 0219        CALL LOADER(FLIP,2)                               02970000
ISN 0220        WRITE (6,901)                                     02980000
ISN 0221        GO TO (340,350,360), NKN                          02990000
ISN 0222    340 WRITE (6,911)  (ROOTR(LL),LL=NL,K1)               03000000
ISN 0223        GO TO 370                                         03010000
ISN 0224    350 WRITE (6,912)  (ROOTR(LL),LL=NL,K1)               03020000
ISN 0225        GO TO 370                                         03030000
ISN 0226    360 WRITE (6,913)  (ROOTR(LL),LL=NL,K1)               03040000
ISN 0227    370 WRITE (6,914)                                     03050000
ISN 0228        DO 380  I=1,N                                     03060000
ISN 0229    380 WRITE(6,915) I,LAP(I),(DARK(I,KK),KK=1,IN)        03070000
ISN 0230        IF(NRMRO.EQ.0) GO TO 387                          03070500
ISN 0232        DO 388 II=1,IN                                    03071000
ISN 0233        IF(ROWRM.GT.XNORET)GO TO 381                      03071500
ISN 0235        ROWN=DARK(NRMRO,II)                               03072000
ISN 0236        GO TO 383                                         03072500
ISN 0237    381 ROWN=DARK(1,II)                                   03073000
ISN 0238        DO 382 M=2,N                                      03073100
ISN 0239    382 IF(ABS(DARK(M,II)).GT.ABS(ROWN)) ROWN=DARK(M,II)  03073400
ISN 0241    383 DO 385 JJ=1,N                                     03073700
ISN 0242    385 DARK(JJ,II)=DARK(JJ,II)/ROWN                      03074000
ISN 0243    386 PHIK(II)=PHIK(II)/ROWN**2                         03074100
ISN 0244        CALL LOADER(FLIP,2)                               03074500
ISN 0245        WRITE(6,901)                                      03074600
ISN 0246        IF(ROWRM.GT.XNORET)GO TO 384                      03074700
ISN 0248        WRITE(6,935) NRMRO                                03075000
ISN 0249        GO TO 389                                         03075500
ISN 0250    384 WRITE(6,936)                                      03075700
ISN 0251    389 WRITE(6,914)                                      03076000
ISN 0252        DO 386 I=1,N                                      03077000
ISN 0253    386 WRITE(6,915) I,LAP(I),(DARK(I,KK),KK=1,IN)        03078000
ISN 0254        CALL LOADER(DUM,2)                                03078100
ISN 0255        DO 395 LLL=1,IN                                   03078200
ISN 0256        SUMA=0.0                                          03078300
ISN 0257        DO 396 J=1,N                                      03078600
ISN 0258        XI=DARK(J,LLL)**2                                 03078900
ISN 0259        DR=XI/(XM(J)*XM(J))                               03079100
ISN 0260    396 SUMA=SUMA+DR                                      03079500
ISN 0261    395 WRITE(6,937) LLL,SUMA,LLL,PHIK(LLL)               03079700
ISN 0262    387 IF(K.NE.NN) GO TO 190                             03080000
```

```
                   C                                                     03090000
                   C                                                     03100000
ISN 0264                 CALL LOADER(FLIP,2)                             03110000
ISN 0265                 WRITE (6,901)                                   03120000
ISN 0266                 WRITE (6,916)                                   03130000
ISN 0267                 DO 400 L =1,N                                   03140000
ISN 0268                 IF (ROOTR(L).GT.0.) GO TO 390                   03150000
                   C                                                     03160000
ISN 0270                 WRITE (6,917) ROOTR(L)                          03170000
ISN 0271                 GO TO 400                                       03180000
ISN 0272             390 SROOT=SQRT(ROOTR(L))                            03190000
ISN 0273                 DSROOT=SROOT/6.28316                            03200000
ISN 0274                 WRITE(6,918) SROOT,ROOTR(L),DSROOT              03210000
ISN 0275             400 CONTINUE                                        03220000
ISN 0276                 IF(EVSAVE.EQ.1.) GO TO 410                      03230000
                   C                                                     03240000
ISN 0278                 NOMASV=NOMASV+1                                 03250000
ISN 0279                 IF (NOMASV.EQ.0) GO TO 440                      03260000
ISN 0281                 REWIND 3                                        03270000
ISN 0282                 REWIND 1                                        03280000
ISN 0283                 WRITE (6,904)                                   03290000
ISN 0284                 GO TO 60                                        03300000
ISN 0285             410 DO 415 I=1,N                                    03303000
ISN 0286                 TUP = XM(I)                                     03307000
ISN 0287                 XM(I) = SCALF / (TUP*TUP)                       03310000
ISN 0288             415 CONTINUE                                        03313000
ISN 0289                 WRITE (3)  MAP,XM,MGB                           03317000
ISN 0290                 DO 420 I=1,512                                  03320000
ISN 0291             420 XM(I) = 500000000                               03330000
ISN 0292                 DO 430 J=1,3                                    03340000
ISN 0293             430 WRITE (3) (XM(I),I=1,512)                       03350000
ISN 0294             440 CONTINUE                                        03360000
ISN 0295                 RETURN                                          03370000
ISN 0296                 END                                             03380000
```

```
LEVEL 21.  ( JAN 73 )                         OS/360  FORTRAN H

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=58,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
               C                                                                 00010000
               C               THIS ROUTINE FORMS THE SYMMETRIC EQUIVALENT       00020000
               C               OF THE DYNAMIC MATRIX,                            00030000
               C                                                                 00040000
               C         MEMBER NAME ST4SJB                                      00050000
               C                       ADDED TO SMI ON 23RD, JAN.,1969           00060000
               C                                                                 00070000
ISN 0002             SUBROUTINE   SJB(XK,XM,WF,N)                                00080000
ISN 0003             IMPLICIT REAL*8(A-H,O-Z)                                    00085000
               C                                                                 00090000
ISN 0004             DIMENSION    XK(1),XM(1)                                    00100000
               C                                                                 00110000
ISN 0005             SQRT(X)=DSQRT(X)                                            00115000
ISN 0006             NN = N*(N+1)/2                                              00120000
ISN 0007             DO 10 I=1,N                                                 00130000
ISN 0008             XM(I) = SQRT(WF/XM(I))                                      00140000
ISN 0009          10 CONTINUE                                                    00150000
ISN 0010             JB = 1                                                      00160000
ISN 0011             IE = 1                                                      00170000
ISN 0012             IS = 1                                                      00180000
ISN 0013             DO 50 K=1,N                                                 00190000
ISN 0014          20 XK(JB) = XM(IS)*XM(IE)*XK(JB)                               00200000
ISN 0015             IF (IS-IE)  30,40,40                                        00210000
ISN 0016          30 IS = IS +1                                                  00220000
ISN 0017             JB = JB +1                                                  00230000
ISN 0018             GO TO 20                                                    00240000
ISN 0019          40 IE = IE +1                                                  00250000
ISN 0020             IS = 1                                                      00260000
ISN 0021             JB = JB +1                                                  00270000
ISN 0022          50 CONTINUE                                                    00280000
ISN 0023             RETURN                                                      00290000
ISN 0024             END                                                         00300000
```

```
LEVEL 21.7   JAN 73 )                                   OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                   C                                                                 00010000
                   C       MEMBER NAME S74LOCI                                       00020000
                   C                                                                 00030000
 ISN 0002                  SUBROUTINE LOCI(I,J,IR,N,M,MS)                            00040000
 ISN 0003                  IMPLICIT REAL*8(A-H,O-Z)                                  00050000
 ISN 0004                  IX=I                                                      00060000
 ISN 0005                  JX=J                                                      00070000
 ISN 0006                  IF(MS-1)10,20,30                                          00080000
 ISN 0007               10 IRX=N*(JX-1)+IX                                           00090000
 ISN 0008                  GO TO 36                                                  00100000
 ISN 0009               20 IF(IX-JX)22,24,24                                         00110000
 ISN 0010               22 IRX=IX+(JX*JX-JX)/2                                       00120000
 ISN 0011                  GO TO 36                                                  00130000
 ISN 0012               24 IRX=JX+(IX*IX-IX)/2                                       00140000
 ISN 0013                  GO TO 36                                                  00150000
 ISN 0014               30 IRX=0                                                     00160000
 ISN 0015                  IF(IX-JX)36,32,36                                         00170000
 ISN 0016               32 IRX=IX                                                    00180000
 ISN 0017               36 IR=IRX                                                    00190000
 ISN 0018                  RETURN                                                    00200000
 ISN 0019                  END                                                       00210000
```

```
LEVEL 21.7   (AN 73 )                                  OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                    C                                                              00010000
                    C     MEMBER NAME S74TPRD                                      00020000
                    C                                                              00030000
  ISN 0002                SUBROUTINE TPRD(A,B,R,N,M,MSA,MSB,L)                     00040000
  ISN 0003                IMPLICIT REAL*8 (A-H,O-Z)                                00050000
  ISN 0004                DIMENSION A(1),B(1),R(1)                                 00060000
  ISN 0005                MS=MSA*10+MSB                                            00070000
  ISN 0006                IF(MS-22)30,10,30                                        00080000
  ISN 0007             10 DO 20 I=1,N                                              00090000
  ISN 0008             20 R(I)=A(I)*B(I)                                           00100000
  ISN 0009                RETURN                                                   00110000
  ISN 0010             30 IR=1                                                     00120000
  ISN 0011                DO 90 K=1,L                                              00130000
  ISN 0012                DO 90 J=1,M                                              00140000
  ISN 0013                R(IR)=0.                                                 00150000
  ISN 0014                DO 80 I=1,N                                              00160000
  ISN 0015                IF(MS)40,60,40                                           00170000
  ISN 0016             40 CALL LOC(I,J,IA,N,M,MSA)                                 00180000
  ISN 0017                CALL LOC(I,K,IB,N,L,MSB)                                 00190000
  ISN 0018                IF(IA)50,80,50                                           00200000
  ISN 0019             50 IF(IB)70,80,70                                           00210000
  ISN 0020             60 IA=N*(J-1)+I                                             00220000
  ISN 0021                IB=N*(K-1)+I                                             00230000
  ISN 0022             70 R(IR)=R(IR)+A(IA)*B(IB)                                  00240000
  ISN 0023             80 CONTINUE                                                 00250000
  ISN 0024             90 IR=IR+1                                                  00260000
  ISN 0025                RETURN                                                   00270000
  ISN 0026                END                                                      00280000
```

```
LEVEL 21.7     AN 73 )                     OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
  ISN 0002            SUBROUTINE  LOC(I,J,IR,N,M,MS)                                00010000
                C     MEMBER NAME S74LOC                                            00010001
                C PURPOSE                                                           00020000
                C   COMPUTE A VECTOR SUBSCRIPT FOR AN ELEMENT IN A MATRIX OF SPECIFIED 00030000
                C   STORAGE MODE                                                    00040000
                C                                                                   00050000
                C DESCRIPTION OF PARAMETERS                                         00060000
                C   I - ROW NUMBER OF ELEMENT                                       00070000
                C   J - COLUMN NUMBER OF ELEMENT                                    00080000
                C   IR- RESULTANT VECTOR SUBSCRIPT                                  00090000
                C   N - NUMBER OF ROWS IN MATRIX                                    00100000
                C   M - NUMBER OF COLUMNS IN MATRIX                                 00110000
                C   MS- ONE DIGIT NUMBER FOR STORAGE MODE OF MATRIX                 00120000
                C         0 - GENERAL                                               00130000
                C         1 - SYMMETRIC                                             00140000
                C         2 - DIAGONAL                                              00150000
                C                                                                   00160000
  ISN 0003            IX = I                                                        00170000
  ISN 0004            JX = J                                                        00180000
  ISN 0005            IF (MS - 1) 10,20,30                                          00190000
  ISN 0006         10 IRX = N * (JX - 1) + IX                                       00200000
  ISN 0007            GO TO 36                                                      00210000
  ISN 0008         20 IF (IX - JX) 22,24,24                                         00220000
  ISN 0009         22 IRX = IX + (JX * JX - JX)/2                                   00230000
  ISN 0010            GO TO 36                                                      00240000
  ISN 0011         24 IRX = JX + (IX * IX - IX)/2                                   00250000
  ISN 0012            GO TO 36                                                      00260000
  ISN 0013         30 IRX = 0                                                       00270000
  ISN 0014            IF (IX - JX) 36,32,36                                         00280000
  ISN 0015         32 IRX = IX                                                      00290000
  ISN 0016         36 IR = IRX                                                      00300000
  ISN 0017            RETURN                                                        00310000
  ISN 0018            END                                                           00320000
```

```
LEVEL 21.1 ( JAN 73 )                OS/360  FORTRAN H

      COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=58,SIZE=0000K,
                         SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                C                                                            00010000
                C     MEMBER NAME S74MXOUT                                   00015000
                C     ...............................................................00020000
                C                                                            00030000
                C        SUBROUTINE MXOUT                                    00040000
                C                                                            00050000
                C        PURPOSE                                             00060000
                C           PRODUCES AN OUTPUT LISTING OF ANY SIZED ARRAY ON 00070000
                C           LOGICAL UNIT 'NOUT'.                             00080000
                C                                                            00090000
                C        USAGE                                               00100000
                C           CALL MXOUT(ICODE,A,N,M,MS,LINS,IPOS,ISP,NOUT)    00110000
                C                                                            00120000
                C        DESCRIPTION OF PARAMETERS                           00130000
                C           ICODE- INPUT CODE NUMBER TO BE PRINTED ON EACH OUTPUT PAGE 00140000
                C           A     - NAME OF OUTPUT MATRIX                    00150000
                C                                                            00160000
                C           N     - NUMBER OF ROWS IN A                      00170000
                C           M     - NUMBER OF COLUMNS IN A                   00180000
                C           MS    - STORAGE MODE OF A WHERE MS =             00190000
                C                       0 = GENERAL                          00200000
                C                       1 = SYMMETRIC                        00210000
                C                       2 = DIAGONAL                         00220000
                C           LINS - NUMBER OF PRINT LINES ON THE PAGE (USUALLY 60) 00230000
                C           IPOS - NUMBER OF PRINT POSITIONS ACROSS THE PAGE 00240000
                C                   (USUALLY 132)                            00250000
                C           ISP  - LINE SPACING CODE, 1 FOR SINGLE SPACE     00260000
                C                                     2 FOR DOUBLE SPACE     00270000
                C           NOUT - LOGICAL UNIT NUMBER FOR OUTPUT            00280000
                C                                                            00290000
                C        REMARKS                                             00300000
                C           NONE                                             00310000
                C                                                            00320000
                C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED       00330000
                C           LOC                                              00340000
                C                                                            00350000
                C        METHOD                                              00360000
                C           THIS SUBROUTINE CREATES A STANDARD OUTPUT LISTING OF ANY  00370000
                C           SIZED ARRAY WITH ANY STORAGE MODE. EACH PAGE IS HEADED WITH00380000
                C           THE CODE NUMBER,DIMENSIONS AND STORAGE MODE OF THE ARRAY,  00390000
                C           EACH COLUMN AND ROW IS ALSO HEADED WITH ITS RESPECTIVE     00400000
                C           NUMBER.                                          00410000
                C                                                            00420000
                C        ...............................................................00430000
                C                                                            00440000
                C                                                            00450000
                C                                                            00460000
                C                                                            00490000
 ISN 0002             SUBROUTINE MXOUT(ICODE,A,N,M,MS,LINS,IPOS,ISP,NOUT)    00500000
 ISN 0003             IMPLICIT REAL*8(A-H,O-Z)                               00505000
```

```
ISN 0004          DIMENSION A(1),B(8)                                                       00510000
              C                                                                             00520000
ISN 0005         901 FORMAT  ( 1H1, 5X, 7HMATRIX ,I5,6X,I3,5H ROWS,6X,I3,                   00530000
                  1             8H COLUMNS,8X,13HSTORAGE MODE ,I1,8X,5HPAGE ,I2,/ )         00540000
ISN 0006         902 FORMAT  ( 12X,8HCOLUMN , 7(3X,I3,10X))                                 00550000
ISN 0007         903 FORMAT  ( 1H  )                                                        00560000
ISN 0008         904 FORMAT  ( 1H , 7X, 4HROW ,I3, 7(E16.6) )                               00570000
ISN 0009         905 FORMAT  ( 1H0, 7X, 4HROW ,I3, 7(E16.6) )                               00580000
ISN 0010         906 FORMAT( ///5X, 7HMATRIX ,I5,6X,I3,5H ROWS,6X,I3,                       00584000
                  1           8H COLUMNS,8X,13HSTORAGE MODE ,I1,8X,5HPAGE ,I2,/ )           00585000
              C                                                                             00590000
ISN 0011          J = 1                                                                     00600000
              C                                                                             00610000
              C        WRITE HEADING                                                        00620000
              C                                                                             00630000
ISN 0012          NEND=IPOS/16 -1                                                           00640000
ISN 0013          LEND=(LINS/ISP)-2                                                         00650000
ISN 0014          IPAGE = 1                                                                 00660000
ISN 0015          WRITE(NOUT,906) ICODE,N,M,MS,IPAGE                                        00663000
ISN 0016          LSTRT = 1                                                                 00666000
ISN 0017          GO TO 25                                                                  00669000
ISN 0018       10 LSTRT = 1                                                                 00670000
ISN 0019       20 WRITE (NOUT,901) ICODE,N,M,MS,IPAGE                                       00680000
ISN 0020       25 JNT = J+NEND-1                                                            00690000
ISN 0021          IPAGE = IPAGE+1                                                           00700000
ISN 0022       30 IF (JNT-M) 50,50,40                                                       00710000
ISN 0023       40 JNT = M                                                                   00720000
ISN 0024       50 CONTINUE                                                                  00730000
ISN 0025          WRITE (NOUT,902)  (JCUR,JCUR=J,JNT)                                       00740000
ISN 0026          IF (ISP-1)  60,60,70                                                      00750000
ISN 0027       60 WRITE (NOUT,903)                                                          00760000
ISN 0028       70 LTEND = LSTRT+LEND-1                                                      00770000
ISN 0029          DO 150 L=LSTRT,LTEND                                                      00780000
              C                                                                             00790000
              C        FORM OUTPUT ROW LINE                                                 00800000
              C                                                                             00810000
ISN 0030          DO 100 K=1,NEND                                                           00820000
ISN 0031          KK = K                                                                    00830000
ISN 0032          JT = J+K-1                                                                00840000
ISN 0033          CALL LOC(L,JT,IJNT,N,M,MS)                                                00850000
ISN 0034          B(K) = 0.0                                                                00860000
ISN 0035          IF (IJNT)  90,90,80                                                       00870000
ISN 0036       80 B(K) = A(IJNT)                                                            00880000
ISN 0037       90 CONTINUE                                                                  00890000
              C                                                                             00900000
              C        CHECK IF LAST COLUMN,  IF YES GO TO 110                              00910000
              C                                                                             00920000
ISN 0038          IF (JT-M) 100,110,110                                                     00930000
ISN 0039      100 CONTINUE                                                                  00940000
              C                                                                             00950000
              C        END OF LINE, NOW WRITE                                               00960000
              C                                                                             00970000
```

```
ISN 0040        110 IF (ISP-1) 120,120,130                                     00980000
ISN 0041        120 WRITE (NOUT,904)  L,(B(JN),JN=1,KK)                        00990000
ISN 0042            GO TO 140                                                  01000000
ISN 0043        130 WRITE (NOUT,905)  L,(S(JN),JN=1,KK)                        01010000
              C                                                                01020000
              C           IF END OF ROWS, GO CHECK COLUMNS                     01030000
              C                                                                01040000
ISN 0044        140 IF (N-L)  160,160,150                                      01050000
ISN 0045        150 CONTINUE                                                   01060000
              C                                                                01070000
              C           END OF PAGE, NOW CHECK FOR MORE OUTPUT               01080000
              C                                                                01090000
ISN 0046            LSTRT = LSTRT+LEND                                         01100000
ISN 0047            GO TO 20                                                   01110000
              C                                                                01120000
              C           END OF COLUMNS, THEN RETURN                          01130000
              C                                                                01140000
ISN 0048        160 IF (JT-M)  170,180,180                                     01150000
ISN 0049        170 J=JT+1                                                     01160000
ISN 0050            GO TO 10                                                   01170000
ISN 0051        180 RETURN                                                     01180000
ISN 0052            END                                                        01190000
```

LEVEL 21.   JAN 73)   OS/360 FORTRAN H

```
COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=58,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
           C
           C    MEMBER NAME S74EIGEN
           C    ..................................................................
           C
           C       SUBROUTINE EIGEN
           C
           C       PURPOSE
           C          COMPUTE EIGENVALUES AND EIGENVECTORS OF A REAL SYMMETRIC
           C          MATRIX
           C
           C       USAGE
           C          CALL EIGEN(A,R,N,MV)
           C
           C       DESCRIPTION OF PARAMETERS
           C          A - ORIGINAL MATRIX (SYMMETRIC), DESTROYED IN COMPUTATION,
           C              RESULTANT EIGENVALUES ARE DEVELOPED IN DIAGONAL OF
           C              MATRIX A IN DESCENDING ORDER.
           C          R - RESULTANT MATRIX OF EIGENVECTORS (STORED COLUMNWISE,
           C              IN SAME SEQUENCE AS EIGENVALUES)
           C          N - ORDER OF MATRICES A AND R
           C          MV- INPUT CODE
           C                  0   COMPUTE EIGENVALUES AND EIGENVECTORS
           C                  1   COMPUTE EIGENVALUES ONLY (R NEED NOT BE
           C                      DIMENSIONED BUT MUST STILL APPEAR IN CALLING
           C                      SEQUENCE)
           C
           C       REMARKS
           C          ORIGINAL MATRIX A MUST BE REAL SYMMETRIC (STORAGE MODE=1)
           C          MATRIX A CANNOT BE IN THE SAME LOCATION AS MATRIX R
           C
           C       SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
           C          NONE
           C
           C       METHOD
           C          DIAGONALIZATION METHOD ORIGINATED BY JACOBI AND ADAPTED
           C          BY VON NEUMANN FOR LARGE COMPUTERS AS FOUND IN =MATHEMATICAL
           C          METHODS FOR DIGITAL COMPUTERS=, EDITED BY A. RALSTON AND
           C          H.S. WILF, JOHN WILEY AND SONS, NEW YORK, 1962, CHAPTER 7
           C
           C    ..................................................................
           C
           C
ISN 0002            SUBROUTINE EIGEN(A,R,N,MV)
ISN 0003            IMPLICIT REAL*8(A-H,O-Z)
ISN 0004            DIMENSION A(1),R(1)
           C
           C          ............................................................
           C
           C       IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE
           C       C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION
```

```
                 C         STATEMENT WHICH FOLLOWS.                                     00510000
                 C                                                                      00520000
                 C      DOUBLE PRECISION A,R,ANORM,ANRMX,THR,X,Y,SINX,SINX2,COSX,       00530000
                 C     1                      COSX2,SINCS                               00540000
                 C                                                                      00550000
                 C         THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS  00560000
                 C         APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS    00570000
                 C         ROUTINE.                                                     00580000
                 C                                                                      00590000
                 C         THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO    00600000
                 C         CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  SQRT IN STATEMENTS00610000
                 C         40, 68, 75, AND 78 MUST BE CHANGED TO DSQRT.  ABS IN STATEMENT 00620000
                 C         62 MUST BE CHANGED TO DABS.                                  00630000
                 C                                                                      00640000
                 C         ...........................................................00650000
                 C                                                                      00660000
                 C         GENERATE IDENTITY MATRIX                                     00670000
                 C                                                                      00680000
ISN 0005                ABS(X)=DABS(X)                                                  00681000
ISN 0006                SQRT(X)=DSQRT(X)                                                00682000
ISN 0007                IF(MV-1) 10,25,10                                               00690000
ISN 0008             10 IQ=-N                                                           00700000
ISN 0009                DO 20 J=1,N                                                     00710000
ISN 0010                IQ=IQ+N                                                         00720000
ISN 0011                DO 20 I=1,N                                                     00730000
ISN 0012                IJ=IQ+I                                                         00740000
ISN 0013                R(IJ)=0.0                                                       00750000
ISN 0014                IF(I-J) 20,15,20                                                00760000
ISN 0015             15 R(IJ)=1.0                                                       00770000
ISN 0016             20 CONTINUE                                                        00780000
                 C                                                                      00790000
                 C         COMPUTE INITIAL AND FINAL NORMS (ANORM AND ANORMX)           00800000
                 C                                                                      00810000
ISN 0017             25 ANORM=0.0                                                       00820000
ISN 0018                DO 35 I=1,N                                                     00830000
ISN 0019                DO 35 J=1,N                                                     00840000
ISN 0020                IF(I-J) 30,35,30                                                00850000
ISN 0021             30 IA=I+(J*J-J)/2                                                  00860000
ISN 0022                ANORM=ANORM+A(IA)*A(IA)                                         00870000
ISN 0023             35 CONTINUE                                                        00880000
ISN 0024                IF(ANORM) 165,165,40                                            00890000
ISN 0025             40 ANORM=1.414*SQRT(ANORM)                                         00900000
ISN 0026                ANRMX=ANORM*1.0D-12/FLOAT(N)                                    00910000
                 C                                                                      00920000
                 C         INITIALIZE INDICATORS AND COMPUTE THRESHOLD, THR             00930000
                 C                                                                      00940000
ISN 0027                IND=0                                                           00950000
ISN 0028                THR=ANORM                                                       00960000
ISN 0029             45 THR=THR/FLOAT(N)                                                00970000
ISN 0030             50 L=1                                                             00980000
ISN 0031             55 M=L+1                                                           00990000
                 C                                                                      01000000
```

```
               C         COMPUTE SIN AND COS                                    01010000
               C                                                                01020000
ISN 0032          60 MQ=(M*M-M)/2                                               01030000
ISN 0033             LQ=(L*L-L)/2                                               01040000
ISN 0034             LM=L+MQ                                                    01050000
ISN 0035          62 IF( ABS(A(LM))-THR) 130,65,65                              01060000
ISN 0036          65 IND=1                                                      01070000
ISN 0037             LL=L+LQ                                                    01080000
ISN 0038             MM=M+MQ                                                    01090000
ISN 0039             X=0.5*(A(LL)-A(MM))                                        01100000
ISN 0040          68 Y=-A(LM)/ SQRT(A(LM)*A(LM)+X*X)                            01110000
ISN 0041             IF(X) 70,75,75                                             01120000
ISN 0042          70 Y=-Y                                                       01130000
ISN 0043          75 SINX=Y/ SQRT(2.0*(1.0+( SQRT(1.0-Y*Y))))                   01140000
ISN 0044             SINX2=SINX*SINX                                            01150000
ISN 0045          78 COSX= SQRT(1.0-SINX2)                                      01160000
ISN 0046             COSX2=COSX*COSX                                            01170000
ISN 0047             SINCS =SINX*COSX                                           01180000
               C                                                                01190000
               C         ROTATE L AND M COLUMNS                                 01200000
               C                                                                01210000
ISN 0048             ILQ=N*(L-1)                                                01220000
ISN 0049             IMQ=N*(M-1)                                                01230000
ISN 0050             DO 125 I=1,N                                               01240000
ISN 0051             IQ=(I*I-I)/2                                               01250000
ISN 0052             IF(I-L) 80,115,80                                          01260000
ISN 0053          80 IF(I-M) 85,115,90                                          01270000
ISN 0054          85 IM=I+MQ                                                    01280000
ISN 0055             GO TO 95                                                   01290000
ISN 0056          90 IM=M+IQ                                                    01300000
ISN 0057          95 IF(I-L) 100,105,105                                        01310000
ISN 0058         100 IL=I+LQ                                                    01320000
ISN 0059             GO TO 110                                                  01330000
ISN 0060         105 IL=L+IQ                                                    01340000
ISN 0061         110 X=A(IL)*COSX-A(IM)*SINX                                    01350000
ISN 0062             A(IM)=A(IL)*SINX+A(IM)*COSX                                01360000
ISN 0063             A(IL)=X                                                    01370000
ISN 0064         115 IF(MV-1) 120,125,120                                       01380000
ISN 0065         120 ILR=ILQ+I                                                  01390000
ISN 0066             IMR=IMQ+I                                                  01400000
ISN 0067             X=R(ILR)*COSX-R(IMR)*SINX                                  01410000
ISN 0068             R(IMR)=R(ILR)*SINX+R(IMR)*COSX                             01420000
ISN 0069             R(ILR)=X                                                   01430000
ISN 0070         125 CONTINUE                                                   01440000
ISN 0071             X=2.0*A(LM)*SINCS                                          01450000
ISN 0072             Y=A(LL)*COSX2+A(MM)*SINX2-X                                01460000
ISN 0073             X=A(LL)*SINX2+A(MM)*COSX2+X                                01470000
ISN 0074             A(LM)=(A(LL)-A(MM))*SINCS+A(LM)*(COSX2-SINX2)              01480000
ISN 0075             A(LL)=Y                                                    01490000
ISN 0076             A(MM)=X                                                    01500000
               C                                                                01510000
               C         TESTS FOR COMPLETION                                   01520000
```

```
                C                                                              01530000
                C        TEST FOR M = LAST COLUMN                              01540000
                C                                                              01550000
ISN 0077          130 IF(M-N) 135,140,135                                      01560000
ISN 0078          135 M=M+1                                                    01570000
ISN 0079              GO TO 60                                                 01580000
                C                                                              01590000
                C        TEST FOR L = SECOND FROM LAST COLUMN                  01600000
                C                                                              01610000
ISN 0080          140 IF(L-(N-1)) 145,150,145                                  01620000
ISN 0081          145 L=L+1                                                    01630000
ISN 0082              GO TO 55                                                 01640000
ISN 0083          150 IF(IND-1) 160,155,160                                    01650000
ISN 0084          155 IND=0                                                    01660000
ISN 0085              GO TO 50                                                 01670000
                C                                                              01680000
                C        COMPARE THRESHOLD WITH FINAL NORM                     01690000
                C                                                              01700000
ISN 0086          160 IF(THR-ANRMX) 165,165,45                                 01710000
                C                                                              01720000
                C        SORT EIGENVALUES AND EIGENVECTORS                     01730000
                C                                                              01740000
ISN 0087          165 IQ=-N                                                    01750000
ISN 0088              DO 185 I=1,N                                             01760000
ISN 0089              IQ=IQ+N                                                  01770000
ISN 0090              LL=I+(I*I-I)/2                                           01780000
ISN 0091              JQ=N*(I-2)                                               01790000
ISN 0092              DO 185 J=I,N                                             01800000
ISN 0093              JQ=JQ+N                                                  01810000
ISN 0094              MM=J+(J*J-J)/2                                           01820000
ISN 0095              IF(A(LL)-A(MM)) 185,185,170                              01830000
ISN 0096          170 X=A(LL)                                                  01840000
ISN 0097              A(LL)=A(MM)                                              01850000
ISN 0098              A(MM)=X                                                  01860000
ISN 0099              IF(MV-1) 175,185,175                                     01870000
ISN 0100          175 DO 180 K=1,N                                             01880000
ISN 0101              ILR=IQ+K                                                 01890000
ISN 0102              IMR=JQ+K                                                 01900000
ISN 0103              X=R(ILR)                                                 01910000
ISN 0104              R(ILR)=R(IMR)                                            01920000
ISN 0105          180 R(IMR)=X                                                 01930000
ISN 0106          185 CONTINUE                                                 01940000
ISN 0107              RETURN                                                   01950000
ISN 0108              END                                                      01960000
```

```
LEVEL 21.   ( JAN 73 )                          OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                    C                                                                  00010000
                    C            A  IS THE SQUARE MATRIX, OF ORDER IO,                 00020000
                    C               TO BE INVERTED,                                    00030000
                    C           IER   IS AN ERROR MESSAGE INDICATOR                    00040000
                    C             0  MEANS 'A' IS SINGULAR                             00050000
                    C             1  MEANS  A NORMAL RETURN                            00060000
                    C             2  MEANS  IO WAS INVALID                             00070000
                    C                                                                  00080000
                    C                                                                  00090000
                    C      MEMBER NAME S74INVA                                         00100000
                    C              ADDED TO SM1 ON 6TH, FEB., 1969                     00110000
                    C                                                                  00120000
  ISN 0002                 SUBROUTINE  INVA(A,IO,IER)                                  00130000
  ISN 0003                 IMPLICIT REAL*8(A-H,O-Z)                                    00135000
                    C                                                                  00140000
  ISN 0004                 DIMENSION A(1),COL(250)                                     00150000
                    C                                                                  00160000
  ISN 0005                 IF (IO-1)  160,150,10                                       00170000
  ISN 0006            10   IF(IO-250) 20,20,160                                        00180000
                    C                                                                  00190000
                    C              ***  INVERSE OF A 2X2 MATRIX                        00200000
                    C                                                                  00210000
  ISN 0007            20   C = A(1)*A(3)-A(2)*A(2)                                     00220000
  ISN 0008                 IF (C) 30,140,30                                            00230000
  ISN 0009            30   A(2) = -A(2)/C                                              00240000
  ISN 0010                 COL(1) = A(1)/C                                             00250000
  ISN 0011                 A(1) = A(3)/C                                               00260000
  ISN 0012                 A(3) = COL(1)                                               00270000
  ISN 0013                 IF (IO-2) 160,120,40                                        00280000
  ISN 0014            40   K = 1                                                       00290000
  ISN 0015                 M = IO-1                                                    00300000
  ISN 0016                 DO 120 L=2,M                                                00310000
  ISN 0017                 K = K+L                                                     00320000
                    C                                                                  00330000
                    C              L,-,H OF SYMMETRIC MATRIX*COLUMN                    00340000
                    C                                                                  00350000
  ISN 0018                 N = 0                                                       00360000
  ISN 0019                 DO 50 I=1,L                                                 00370000
  ISN 0020            50   COL (I) = 0                                                 00380000
  ISN 0021                 DO 70 I=1,L                                                 00390000
  ISN 0022                 IA = K+I                                                    00400000
  ISN 0023                 DO 70 J=1,I                                                 00410000
  ISN 0024                 N = N+1                                                     00420000
  ISN 0025                 COL(J)=COL(J)+A(N)*A(IA)                                    00430000
  ISN 0026                 IF (J-I) 60,70,160                                          00440000
  ISN 0027            60   IB = K+J                                                    00450000
  ISN 0028                 COL(I)=COL(I)+A(N)*A(IB)                                    00460000
  ISN 0029            70   CONTINUE                                                    00470000
                    C                                                                  00480000
                    C              COMPUTE    B22                                      00490000
```

```
                  C                                         00500000
ISN 0030                C  = 0                              00510000
ISN 0031                DO 80 I=1,L                         00520000
ISN 0032                IA = K+I                            00530000
ISN 0033          80    C = C+ A(IA)*COL(I)                 00540000
ISN 0034                IA = IA+1                           00550000
ISN 0035                C = A(IA)-C                         00560000
ISN 0036                IF (C) 90,140,90                    00570000
ISN 0037          90    C = 1.0/C                           00580000
ISN 0038                A(IA) = C                           00590000
                  C                                         00600000
                  C             COMPUTE   B21               00610000
                  C                                         00620000
ISN 0039                DO 100 I=1,L                        00630000
ISN 0040                IA = K+I                            00640000
ISN 0041         100    A(IA) = -C*COL(I)                   00650000
                  C                                         00660000
                  C             COMPUTE   B11               00670000
                  C                                         00680000
ISN 0042                N  = 0                              00690000
ISN 0043                DO 110 I=1,L                        00700000
ISN 0044                DO 110 J=1,I                        00710000
ISN 0045                N  = N +1                           00720000
ISN 0046                IA = K+J                            00730000
ISN 0047         110    A(N)=A(N)-A(IA)*COL(I)              00740000
ISN 0048         120    CONTINUE                            00750000
ISN 0049         130    IER = 1                             00760000
ISN 0050                RETURN                              00770000
ISN 0051         140    IER = 0                             00780000
ISN 0052                RETURN                              00790000
ISN 0053         150    A(1) = 1.0/A(1)                     00800000
ISN 0054                GO TO 130                           00810000
ISN 0055         160    IER = 2                             00820000
ISN 0056                RETURN                              00830000
ISN 0057                END                                 00840000
```

```
LEVEL 21.     JUL 73 )                          OS/360  FORTRAN H

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                C                                                                  00010000
                C      MEMBER NAME S74GELS                                          00015000
                C                                                                  00020000
                C              SUBROUTINE  GELS  -  USED TO SOLVE A SYSTEM OF      00030000
                C              SIMULTANEOUS LINEAR EQUATIONS WITH SYMMETRIC        00040000
                C              COEFFICIENT MATRIX WHOSE UPPER TRIANGULAR          00050000
                C              PART IS ASSUMED TO BE STORED COLUMNWISE.           00060000
                C                                                                  00070000
                C                                                                  00100000
 ISN 0002              SUBROUTINE GELS(R,A,M,N,EPS,IER,AUX)                        00110000
 ISN 0003              IMPLICIT REAL*8(A-H,O-Z)                                    00115000
                C                                                                  00120000
 ISN 0004              DIMENSION A(1),R(1),AUX(1)                                  00130000
                C                                                                  00140000
 ISN 0005              ABS(X)=DABS(X)                                              00145000
 ISN 0006              EPS=1.0D-16                                                 00145500
 ISN 0007              IF(M)24,24,1                                                00150000
 ISN 0008            1 IER=0                                                       00160000
 ISN 0009              PIV=0.D0                                                    00170000
 ISN 0010              L=0                                                         00180000
 ISN 0011              DO 3 K=1,M                                                  00190000
 ISN 0012              L=L+K                                                       00200000
 ISN 0013              TB=ABS(A(L))                                                00210000
 ISN 0014              IF(TB-PIV)3,3,2                                             00220000
 ISN 0015            2 PIV=TB                                                      00230000
 ISN 0016              I=L                                                         00240000
 ISN 0017              J=K                                                         00250000
 ISN 0018            3 CONTINUE                                                    00260000
 ISN 0019              TOL=EPS*PIV                                                 00270000
 ISN 0020              LST=0                                                       00280000
 ISN 0021              NM=N*M                                                      00290000
 ISN 0022              LEND=M-1                                                    00300000
 ISN 0023              DO 18 K=1,M                                                 00310000
 ISN 0024              IF(PIV)24,24,4                                              00320000
 ISN 0025            4 IF(IER)7,5,7                                                00330000
 ISN 0026            5 IF(PIV-TOL)6,6,7                                            00340000
 ISN 0027            6 IER=K-1                                                     00350000
 ISN 0028            7 LT=J-K                                                      00360000
 ISN 0029              LST=LST+K                                                   00370000
 ISN 0030              PIVI=1.D0 /A(I)                                             00380000
 ISN 0031              DO 8 L=K,NM,M                                               00390000
 ISN 0032              LL=L+LT                                                     00400000
 ISN 0033              TB=PIVI*R(LL)                                               00410000
 ISN 0034              R(LL)=R(L)                                                  00420000
 ISN 0035            8 R(L)=TB                                                     00430000
 ISN 0036              IF(K-M)9,19,19                                              00440000
 ISN 0037            9 LR=LST+(LT*(K+J-1))/2                                       00450000
 ISN 0038              LL=LR                                                       00460000
 ISN 0039              L=LST                                                       00470000
 ISN 0040              DO 14 II=K,LEND                                             00480000
```

```
      L=L+II
      LL=LL+1
      IF(L-LR)12,10,11
   10 A(LL)=A(LST)
      TB=A(L)
      GO TO 13
   11 LL=L+LT
   12 TB=A(LL)
      A(LL)=A(L)
   13 AUX(II)=TB
   14 A(L)=PIVI*TB
      A(LST)=LT
      PIV = 0.D0
      LLST=LST
      LT=0
      DO 18 II=K,LEND
      PIVI=-AUX(II)
      LL=LLST
      LT=LT+1
      DO 15 LLD=II,LEND
      LL=LL+LLD
      L=LL+LT
   15 A(L)=A(L)+PIVI*A(LL)
      LLST=LLST+II
      LR=LLST+LT
      TB=ABS(A(LR))
      IF(TB-PIV)17,17,16
   16 PIV=TB
      I=LR
      J=II+1
   17 DO 18 LR=K,NM,M
      LL=LR+LT
   18 R(LL)=R(LL)+PIVI*R(LR)
   19 IF(LEND)24,23,20
   20 II=M
      DO 22 I=2,M
      LST=LST-II
      II=II-1
      L=A(LST)+.5D0
      DO 22 J=II,NM,M
      TB=R(J)
      LL=J
      K=LST
      DO 21 LT=II,LEND
      LL=LL+1
      K=K+LT
   21 TB=TB-A(K)*R(LL)
      K=J+L
      R(J)=R(K)
   22 R(K)=TB
   23 RETURN
   24 IER=-1
```

```
ISN 0093          RETURN                                                    01010000
ISN 0094          END                                                       01020000

LEVEL 21.7 ( JAN 73 )                      OS/360  FORTRAN H

       COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                          SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                C     MEMBER NAME S74DPLDR                                  00010000
                C                                                           00015000
  ISN 0002            SUBROUTINE DLOADR(ARRAY,MODE)                         00020000
  ISN 0003            IMPLICIT REAL*8(A-H,O-S,U-Z)                          00025000
  ISN 0004            DIMENSION  CARD(5),TITLE(19),ARRAY(1)                 00030000
  ISN 0005            GO TO (100,200),MODE                                  00040000
  ISN 0006      100   CONTINUE                                              00050000
  ISN 0007      105   READ(5,101)NSUB,NOPC,CARD                             00060000
  ISN 0008      101   FORMAT(I4,I1,5E14.7)                                  00070000
  ISN 0009            IF(NSUB)303,303,106                                   00080000
  ISN 0010      106   IF(NOPC)303,303,107                                   00090000
  ISN 0011      107   IF(NOPC-6)109,303,108                                 00100000
  ISN 0012      108   IF(NOPC-8)102,111,112                                 00110000
  ISN 0013      109   DO 110  JCARD=1,NOPC                                  00120000
  ISN 0014            JARRAY=NSUB+JCARD-1                                   00130000
  ISN 0015      110   ARRAY(JARRAY)=CARD(JCARD)                             00140000
  ISN 0016            GO TO 105                                             00150000
  ISN 0017      102   READ(5,103)TITLE                                      00160000
  ISN 0018      103   FORMAT(16,17A4,A2)                                    00170000
  ISN 0019            IF(TITLE(1))301,104,301                               00180000
  ISN 0020      104   NPAGE=0                                               00190000
  ISN 0021            GO TO 105                                             00200000
  ISN 0022      111   RETURN                                                00210000
  ISN 0023      112   STOP                                                  00220000
  ISN 0024      200   NPAGE=NPAGE+1                                         00230000
  ISN 0025            WRITE(6,201)(TITLE(N),N=2,19),NPAGE                   00240000
  ISN 0026      201   FORMAT(1H1,17A4,A2,6H PAGE ,I3,//)                    00250000
  ISN 0027            RETURN                                                00260000
  ISN 0028      301   WRITE(6,302)                                          00270000
  ISN 0029      302   FORMAT(20H INVALID HEADER CARD)                       00280000
  ISN 0030            STOP                                                  00290000
  ISN 0031      303   WRITE(6,304)NSUB,NOPC,ARRAY                           00300000
  ISN 0032      304   FORMAT(19H INVALID DATA CARD ,I4,I1,5E14.7)           00310000
  ISN 0033            STOP                                                  00320000
  ISN 0034            END                                                   00330000
```

```
          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                C                                                              00010000
                C      MEMBER NAME S74MTPR                                     00020000
                C                                                              00030000
                C           MATRIX PRINT-OUT SUBROUTINE  - CF, D96MATPR        00040000
                C                                                              00050000
ISN 0002               SUBROUTINE MATPRN (A,NTR,NTC,MR,MC)                     00060000
ISN 0003               IMPLICIT REAL*8(A-H,O-Z)                                00065000
                C                                                              00070000
ISN 0004               DIMENSION  A(1)                                         00080000
                C                                                              00090000
ISN 0005             1 FORMAT   (8HOCOLUMNS  I8, 4I16 77H     ROW)             00100000
ISN 0006             2 FORMAT   (1H ,I7, 5E15.8)                               00110000
ISN 0007             3 FORMAT(1H1,                                             00120000
                      1/ 37X,'(S-74 PROGRAM)')                                 00130000
ISN 0008               ICOS=0                                                  00131000
ISN 0009               NCOS=5                                                  00140000
ISN 0010               NR = 1                                                  00150000
ISN 0011               NC = 5                                                  00160000
ISN 0012               NOUT = 6                                                00170000
ISN 0013            10 IF (NC.LE.NTC) GO TO 11                                 00180000
ISN 0015               NCOS=NTC-NCOS*ICOS                                      00190000
ISN 0016               NC=NTC                                                  00194000
ISN 0017               NOUT = 6                                                00195000
ISN 0018            11 WRITE (NOUT,1)  (JBO,JBO=NR,NC)                         00200000
ISN 0019               ICOS=ICOS+1                                             00201000
ISN 0020               KOUNT = 0                                               00210000
ISN 0021               K1=MR*NC                                                00220000
ISN 0022               MCOS=NCOS*MR                                            00221000
ISN 0023               DO 13 I=1,NTR                                           00230000
ISN 0024               IF (KOUNT.LE.40) GO TO 12                               00240000
ISN 0026               WRITE (NOUT,3)                                          00250000
ISN 0027               WRITE (NOUT,1)  (JBO,JBO=NR,NC)                         00260000
ISN 0028               KOUNT=0                                                 00310100
ISN 0029            12 KI=K1-MCOS+I                                            00310200
ISN 0030               WRITE(NOUT,2) I,(A(K),K=KI,K1,MR)                       00310300
ISN 0031            13 KOUNT=KOUNT+1                                           00310400
                C                                                              00320000
ISN 0032               IF (NC.EQ.NTC)  GO TO 14                                00330000
ISN 0034               NR = NR +5                                              00340000
ISN 0035               NC = NC +5                                              00350000
ISN 0036               WRITE (NOUT,3)                                          00360000
ISN 0037               GO TO 10                                                00370000
                C                                                              00380000
ISN 0038            14 RETURN                                                  00390000
ISN 0039               END                                                     00400000
```

```
LEVEL 21.7   JAN 73 )                         OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=50,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                  C                                                              00010000
                  C              THIS ROUTINE COPIES THE TAPE PRODUCED BY DR2 IN 00020000
                  C              ABLNK3 USING IBCOM1 ONTO ANOTHER TAPE IN A FORM 00030000
                  C              SUITABLE FOR DIRECT INPUT INTO CSILK4 & DB5,    00040000
                  C                                                              00050000
                  C      MEMBER NAME S74TPC                                      00060000
                  C                 ADDED TO SM1 ON 13TH, MARCH,1969             00070000
                  C                                                              00080000
 ISN 0002               SUBROUTINE  TPCOPY(M,N,NTP,MSO)                          00090000
 ISN 0003               IMPLICIT REAL*8(A-H,O-Z)                                 00095000
 ISN 0004               REAL*4 EBS,SBRS,TGYS                                     00096000
                  C                                                              00100000
 ISN 0005               DIMENSION DAR(250),FCT(250,50),LMN(250),SBR(250),TGY(30),00110000
                       1          EBS(250),SBRS(250),TGYS(30)                    00111000
                  C                                                              00120000
 ISN 0006               REWIND  NTP                                              00140000
 ISN 0007               DO 20 IA=1,N                                             00150000
 ISN 0008               READ (NTP)  (DAR(IC),IC=1,M)                             00160000
 ISN 0009               DO 10 IB=1,M                                             00170000
 ISN 0010            10 FCT(IB,IA)  =  DAR(IB)                                   00180000
 ISN 0011            20 CONTINUE                                                 00190000
 ISN 0012               READ (NTP)  LMN,SBR,TGY                                  00200000
 ISN 0013               REWIND  MSO                                              00220000
 ISN 0014               REWIND 3                                                 00225000
 ISN 0015               WRITE (MSO)  M,N                                         00230000
 ISN 0016               WRITE(3)M,N                                              00235000
 ISN 0017               DO 40 IA=1,N                                             00240000
 ISN 0018               DO 30 IB=1,M                                             00250000
 ISN 0019               EBS(IB)=FCT(IB,IA)                                       00255000
 ISN 0020            30 DAR(IB) = FCT(IB,IA)                                     00260000
 ISN 0021               WRITE(3) (EBS(IC),IC=1,M)                                00265000
 ISN 0022               WRITE (MSO)  (DAR(IC),IC=1,M)                            00270000
 ISN 0023            40 CONTINUE                                                 00280000
 ISN 0024               WRITE (MSO)  LMN                                         00290000
 ISN 0025               WRITE (MSO)  SBR                                         00300000
 ISN 0026               WRITE (MSO)  TGY                                         00310000
 ISN 0027               WRITE(3) LMN                                             00312000
 ISN 0028               DO 50 I=1,250                                            00314000
 ISN 0029            50 SBRS(I)=SBR(I)                                           00314100
 ISN 0030               DO 51 I=1,30                                             00315000
 ISN 0031            51 TGYS(I)=TGY(I)                                           00317000
 ISN 0032               WRITE(3) SBRS                                            00318000
 ISN 0033               WRITE(3) TGYS                                            00319000
 ISN 0034               END FILE MSO                                             00320000
 ISN 0035               END FILE 3                                               00325000
 ISN 0036               REWIND MSO                                               00330000
 ISN 0037               REWIND 3                                                 00335000
 ISN 0038               RETURN                                                   00340000
 ISN 0039               END                                                      00350000
```

```
LEVEL 21.7    AN 73 )                                  OS/360  FORTRAN H

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=54,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,ID,NOXREF
                  C                                                            00010000
                  C     MEMBER NAME S74A6LNS                                   00020000
                  C                                                            00030000
  ISN 0002              SUBROUTINE A6LNKS                                      00040000
  ISN 0003              IMPLICIT REAL*8(A-H,O-Z)                               00050000
  ISN 0004              REAL*4 RS,RC                                           00060000
  ISN 0005              COMMON/IKO/IKO,KEY                                     00070000
  ISN 0006              COMMON/LINKA1/NTOTAL,NORET,NTAPE                       00080000
  ISN 0007              DIMENSION XK(23005)                                    00090000
  ISN 0008              DIMENSION AAA(215),BBB(215),R(215),RS(215,20),RC(215,20)  00100000
  ISN 0009              REWIND 12                                              00110000
  ISN 0010              DO 10 I=1,NORET                                        00120000
  ISN 0011              J=NORET-I+1                                            00130000
  ISN 0012              J2=J*(J+1)/2                                           00140000
  ISN 0013              J1=J2-J+1                                              00150000
  ISN 0014           10 READ(12)(XK(J),J=J1,J2)                                00160000
  ISN 0015              DO 20 I=1,IKO                                          00170000
  ISN 0016              READ(12)((AAA(J),J=1,NORET),(BBB(J),J=1,NORET))        00180000
  ISN 0017              CALL TPRD(XK,AAA,R,NORET,NORET,1,0,1)                  00190000
  ISN 0018              DO 30 K=1,NORET                                        00200000
  ISN 0019           30 RS(K,I)=R(K)                                           00210000
  ISN 0020              CALL TPRD(XK,BBB,R,NORET,NORET,1,0,1)                  00220000
  ISN 0021              DO 40 K=1,NORET                                        00230000
  ISN 0022           40 RC(K,I)=R(K)                                           00240000
  ISN 0023              WRITE(6,300)I                                          00250000
  ISN 0024          300 FORMAT(1H1,24X,'ARD-S-74'//19X,'VIBRATORY INTERNAL LOADS',  00260000
                       1//,27X,'CASE ',I2)                                     00270000
  ISN 0025              WRITE(6,100)                                           00280000
  ISN 0026          100 FORMAT(//,6X,'SINE',14X,'SINE',13X,'COSINE',12X,'COSINE',  00290000
                       1/,3X,'DEFLECTIONS',10X,'LOADS',10X,'DEFLECTIONS',10X,  00300000
                       2'LOADS',/,5X,'(INCHES)',7X,'LBS & IN-LBS',9X,          00310000
                       3'(INCHES)',7X,'LBS & IN-LBS',/,7X,'(A)',14X,           00320000
                       4'(RS)',14X,'(B)',14X,'(RC)',/)                         00330000
  ISN 0027              WRITE(6,200)(AAA(L),RS(L,I),BBB(L),RC(L,I),L=1,NORET)  00340000
  ISN 0028          200 FORMAT(1X,F13.8,5X,F13.4,5X,F13.8,5X,F13.4)            00350000
  ISN 0029           20 CONTINUE                                               00360000
  ISN 0030              REWIND 12                                              00370000
  ISN 0031              IKO2=IKO*2                                             00380000
  ISN 0032              IF(KEY.EQ.1) IKO2=-IKO2                                00380100
  ISN 0034              WRITE(12)NORET,IKO2                                    00390000
  ISN 0035              DO 50 I=1,IKO                                          00400000
  ISN 0036              WRITE(12)(RS(J,I),J=1,NORET)                           00411000
  ISN 0037           50 WRITE(12)(RC(J,I),J=1,NORET)                           00411200
  ISN 0038              REWIND 12                                              00420000
  ISN 0039              RETURN                                                 00430000
  ISN 0040              END                                                    00440000
```

APPENDIX F

LISTING OF COMPUTER PROGRAM FOR USE WITH NASTRAN, LEVEL 16, TO TAKE PUNCHED STRAIN ENERGY OUTPUT FROM RIGID FORMAT 1 TO SORT AND ANALYZE

```
FORTRAN IV G LEVEL  21              MAIN              DATE = 77245          12/17/20              PAGE 0001

            DIMENSION CARD(20),CARD1(20),CARD2(20),CARD3(20),CARD4(20),
           1          CARD5(20),CARD6(20),IM(10000),AAA(10000),BBB(10000)
            DIMENSION SCARD1(20),SCARD2(20),SCARD3(20),SCARD4(20),
           1          SCARD5(20),SCARD6(20)
            DIMENSION FILLER(4)
            DATA TEST /4HSTIT/
      360   FORMAT(/1X,84HINDEX        ELEMENT        STRAIN          PERCENT
          1        ACCUMULATED
          2       /1X, 7HH NO.           NO.
          3        PERCENT    /  )
            IS = 0
            ICASE = 0
            IK = 1
            KK = 11
      5     READ(5,10,END=80) CARD
      10    FORMAT(20A4)
            IF (CARD(1).EQ.TEST) GO TO 25
            READ(0,20) IM(IK),AAA(IK),BBB(IK)
      20    FORMAT(I10,13X,1PE13.5,5X,1PE13.5)
            IF (AAA(IK).LT.0.0) AAA(IK) = 0.0
            IF (BBB(IK).LT.0.0) BBB(IK) = 0.0
            IK = IK + 1
            IF (IK.GT.10000) GO TO 30
            GO TO 5
      25    READ(0,10) CARD1
            READ(5,10) CARD2
            READ(5,10) CARD3
            READ(5,10) CARD4
            READ(5,10) CARD5
            READ(5,10) CARD6
            READ(0,15) FILLER,IC
      15    FORMAT(4A4,I9)
            IF (ICASE.NE.0) GO TO 35
      18    WRITE(6,7)
      7     FORMAT(1H1)
            WRITE(6,10) CARD1
            WRITE(6,10) CARD2
            WRITE(6,10) CARD3
            WRITE(6,10) CARD4
            WRITE(6,10) CARD5
            WRITE(6,10) CARD6
            WRITE(6,390)
            DO 19 I=1,20
            SCARD1(I) = CARD1(I)
            SCARD2(I) = CARD2(I)
            SCARD3(I) = CARD3(I)
            SCARD4(I) = CARD4(I)
            SCARD5(I) = CARD5(I)
      19    SCARD6(I) = CARD6(I)
            ICASE = IC
            GO TO 5
      35    IF (ICASE.EQ.IC) GO TO 5
```

```
FORTRAN IV G LEVEL  21                MAIN           DATE = 77245        12/17/23          PAGE 0002

         40   IND = 0
              IK1 = IK - 2
              DO 50 I=1,IK1
              IF (BBB(I).GE.BBB(I+1)) GO TO 50
              IND = 1
              SWAP1 = AAA(I)
              SWAP2 = BBB(I)
              ISWAP = IM(I)
              AAA(I) = AAA(I+1)
              BBB(I) = BBB(I+1)
              IM(I) = IM(I+1)
              AAA(I+1) = SWAP1
              BBB(I+1) = SWAP2
              IM(I+1) = ISWAP
         50   CONTINUE
              IF (IND.EQ.1) GO TO 40
              SUM = 0.0
              IKK = IK - 1
              DO 60 J=1,IKK
              SUM = SUM + BBB(J)
              WRITE(6,70) J,IM(J),AAA(J),BBB(J),SUM
         70   FORMAT(1X,I5,5X,I9,5X,F10.5,7X,F10.5,7X,F10.5)
              KK = KK + 1
              IF (KK.LT.61) GO TO 60
              WRITE(6,10) SCARD1
              WRITE(6,10) SCARD2
              WRITE(6,10) SCARD3
              WRITE(6,10) SCARD4
              WRITE(6,10) SCARD5
              WRITE(6,10) SCARD6
              WRITE(6,300)
              KK = 10
         60   CONTINUE
              IF (IS.EQ.1) GO TO 100
              KK = 10
              IK = 1
              GO TO 18
         30   WRITE(6,45)
         45   FORMAT(10X,'GREATER THAN 10000 CARDS WERE INPUT JOB ABORTED'  )
         80   IS = 1
              GO TO 40
        100   CALL EXIT
              END
```

APPENDIX G

LISTING OF COMPUTER PROGRAM FOR USE WITH NASTRAN, LEVEL 16, TO REFORMAT RIGID FORMAT 3 PUNCHED EIGENVECTOR OUTPUT, FOR EXECUTION WITH RIGID FORMAT 1 AS SPC CARDS STRAIN ENERGY CALCULATIONS

```
      DIMENSION CARD(18),COM(6)
      DIMENSION FILLER(4),G(2)
      DIMENSION CONT(2)
      DIMENSION SPC(2),DUM(2)
      DATA SPCS,SPCD,ASTER,BLANK /4HSPC*,4HSPCD,4H*   ,4H
      DATA BAD /4HBAD /
      DATA TEST1 /4HSUBC/
      DATA TEST2 /4HTITL/
      IRK = 0
      SPC(1) = SPCS
      SPC(2) = BLANK
      KOUNT = 0
      REWIND 8
    5 READ(5,10,END=100) CARD,INTX
   10 FORMAT(18A4,I8)
      KOUNT = KOUNT + 1
   12 IF (INTX.NE.KOUNT) GO TO 60
      IF (CARD(1).NE.TEST1) GO TO 5
      READ(0,15) FILLER,ID
   15 FORMAT(4A4,I8)
      READ(5,10) CARD,INTX
      KOUNT = KOUNT + 1
   17 READ(5,10,END=100) CARD,INTX
      KOUNT = KOUNT + 1
      IF (INTX.NE.KOUNT) GO TO 70
   19 READ(0,20) IGRID,G,COM(1),COM(2),COM(3),INTX
   20 FORMAT(I10,2A4,5X,1PE13.6,5X,1PE13.6,5X,1PE13.6,I8)
   33 READ(5,10,END=100) CARD,INTX
      KOUNT = KOUNT + 1
      IF (INTX.NE.KOUNT) GO TO 27
      READ(0,25) CONT,COM(4),COM(5),COM(6),INTX
   25 FORMAT(2A4,15X,1PE13.6,5X,1PE13.6,5X,1PE13.6,I8)
   37 DO 40 I=1,6
      IF (COM(I).NE.0.0) WRITE(6,30) SPC,ID,IGRID,I,COM(I)
      IF (COM(I).NE.0.0) WRITE(7,21) SPC,ID,IGRID,I,COM(I)
      IF (COM(I).NE.0.0.AND.IRK.EQ.0) WRITE(8,21) SPC,ID,IGRID,I,COM(I)
   30 FORMAT(1X,2A4,I4,12X,I4,12X,I1,15X,1PE13.6)
   21 FORMAT(2A4,I4,12X,I4,12X,I1,15X,1PE13.6)
   40 CONTINUE
      READ(5,10,END=100) CARD,INTX
      KOUNT = KOUNT + 1
      IF (INTX.NE.KOUNT) GO TO 70
      IF (CARD(1).EQ.TEST2.AND.IRK.EQ.0) GO TO 50
      IF (CARD(1).EQ.TEST2) GO TO 5
      GO TO 19
```

```
60    WRITE(6,80) CARD,INTX
80    FORMAT(10X,'THIS CARD OUT OF SEQUENCE ',18A4,I8)
      KOUNT = KOUNT + 1
      GO TO 5
70    WRITE(6,80) CARD,INTX
      DO 83 I=1,6
      COM(I) = BAD
      WRITE(6,32) SPC,ID,IGRID,I,COM(I)
      WRITE(7,65) SPC,ID,IGRID,I,COM(I)
83    CONTINUE
      KOUNT = KOUNT + 1
      GO TO 17
27    READ(0,20) IGRD, G,COM1,COM2,COM3,INTX
      WRITE(6,80) CARD,INTX
      COM(4) = BAD
      COM(5) = BAD
      COM(6) = BAD
      DO 72 I=1,3
      IF (COM(I).NE.0.0) WRITE(6,30) SPC,ID,IGRID,I,COM(I)
72    CONTINUE
      DO 82 I=4,6
      WRITE(6,32) SPC,ID,IGRID,I,COM(I)
32    FORMAT(1X,2A4,I4,12X,I4,12X,I1,15X,A4)
      WRITE(7,65) SPC,ID,IGRID,I,COM(I)
65    FORMAT(2A4,I4,12X,I4,12X,I1,15X,1PE13.6)
82    CONTINUE
      IGRID = IGRD
      COM(1) = COM1
      COM(2) = COM2
      COM(3) = COM3
      KOUNT = KOUNT + 1
      GO TO 33
50    SPC(1) = SPCD
      SPC(2) = ASTER
      END FILE 8
      REWIND 8
      IRK = 1
55    READ(8,21,END=5) DUM,ID,IGRID,I,COMA
      WRITE(6,30) SPC,ID,IGRID,I,COMA
      WRITE(7,21) SPC,ID,IGRID,I,COMA
      GO TO 55
100   CALL EXIT
      END
```

APPENDIX H

LISTING OF PROGRAM TO IDENTIFY AREAS OF HIGH STRAIN ENERGY DENSITY FOR SHAFTING (S-68)

```
MEMBER NAME  AXIAL601
      SUBROUTINE AXIAL
      COMMON /BLOCK5/  ITEMP(68)
      DIMENSION TEMP(68)
      COMMON/BLOCK2/            NX(24),STRESS(24,24),STIFFK(24,24),
     1U1(24,24),U2(24,24),U3(24,24),U4(24,24),U5(24,24),U6(24,24),
     2U7(24,24),NODE(4)
      DIMENSION INDEX(4)
      EQUIVALENCE (TEMP(1),ITEMP(1))
      NODE(1)=ITEMP(2)
      NODE(2)=ITEMP(3)
      AXAREA=TEMP(4)
      EE    =TEMP(5)
      X1    =TEMP(6)
      Y1    =TEMP(7)
      Z1    =TEMP(8)
      X2    =TEMP(9)
      Y2    =TEMP(10)
      Z2    =TEMP(11)
      CALL UNPAK2(ITEMP(12), NX(1))
      CALL UNPAK2(ITEMP(13), NX(7))
      INDEX(1)=ITEMP(14)
      INDEX(2)=ITEMP(15)
      DO 3 J=1,12
      DO 3 I=1,12
      STRESS(I,J)=0.0
    3 STIFFK(I,J)=0.
      ALENTH=SQRT((X1-X2)**2+(Y1-Y2)**2+(Z1-Z2)**2)
      ELEMK=AXAREA*EE/ALENTH
      VOLUME = ALENTH * AXAREA
      DCXX=(X2-X1)/ALENTH
      DCYX=(Y2-Y1)/ALENTH
      DCZX=(Z2-Z1)/ALENTH
      A=ELEMK*DCXX**2
      B=ELEMK*DCXX*DCYX
      C=ELEMK*DCXX*DCZX
      D=ELEMK*DCYX**2
      E=ELEMK*DCYX*DCZX
      F=ELEMK*DCZX**2
      CONST = SQRT((AXAREA * ALENTH)/(2.* EE))
      STRESS(1,10) = (ELEMK * DCXX / AXAREA) * CONST
      STRESS(1,11) = (ELEMK * DCYX / AXAREA) * CONST
      STRESS(1,12) = (ELEMK * DCZX / AXAREA) * CONST
      STRESS(1,4)=-STRESS(1,10)
      STRESS(1,5)=-STRESS(1,11)
      STRESS(1,6)=-STRESS(1,12)
      STIFFK(4,4)=A
      STIFFK(4,5)=B
      STIFFK(4,6)=C
      STIFFK(4,10)=-A
      STIFFK(4,11)=-B
      STIFFK(4,12)=-C
      STIFFK(5,4)=B
      STIFFK(5,5)=D
      STIFFK(5,6)=E
      STIFFK(5,10)=-B
      STIFFK(5,11)=-D
      STIFFK(5,12)=-E
      STIFFK(6,4)=C
```

```
MEMBER NAME  AXIAL681
      STIFFK(6,5)=E                                                     00800000
      STIFFK(6,6)=F                                                     00810000
      STIFFK(6,10)=-C                                                   00820000
      STIFFK(6,11)=-E                                                   00830000
      STIFFK(6,12)=-F                                                   00840000
      STIFFK(10,4)=-A                                                   00850000
      STIFFK(10,5)=-B                                                   00860000
      STIFFK(10,6)=-C                                                   00870000
      STIFFK(10,10)=A                                                   00880000
      STIFFK(10,11)=B                                                   00890000
      STIFFK(10,12)=C                                                   00900000
      STIFFK(11,4)=-B                                                   00910000
      STIFFK(11,5)=-D                                                   00920000
      STIFFK(11,6)=-E                                                   00930000
      STIFFK(11,10)=B                                                   00940000
      STIFFK(11,11)=D                                                   00950000
      STIFFK(11,12)=E                                                   00960000
      STIFFK(12,4)=-C                                                   00970000
      STIFFK(12,5)=-E                                                   00980000
      STIFFK(12,6)=-F                                                   00990000
      STIFFK(12,10)=C                                                   01000000
      STIFFK(12,11)=E                                                   01010000
      STIFFK(12,12)=F                                                   01020000
      CALL DANDS(STIFFK,STRESS,12,1,NX,INDEX,NODE,VOLUME)               01050000
      RETURN                                                            01060000
      END                                                               01070000

MEMBER NAME  BEAM681
      SUBROUTINE BEAM                                                   00010000
      COMMON /BLOCK5/  ITEMP(68)                                        00060000
      COMMON/BLOCK2/          NX(24),STRESS(24,24),STIFFK(24,24),       00070000
     1AMFLAT(24,24),AMFDA(24,24),U1(24,24),U2(24,24),U3(24,24),U4(24,24)00080000
     2,U5(24,24),NODE(4)                                                00090000
      COMMON/COMSD/ALENTH,E,BIYY,AYY,G,BIZZ,AZZ,NFIX1,NFIX2,NFIX3,      00100000
     1NFIX4,NFIX5,NFIX6,BJ,AKAREA                                       00110000
      DIMENSION TEMP(68)                                                00120000
      DIMENSION INDEX(4)                                                00130000
      EQUIVALENCE (TEMP(1),ITEMP(1))                                    00140000
      DATA IBLANK /1H /                                                 00150000
      NODE(1)=ITEMP(2)                                                  00180000
      NODE(2)=ITEMP(3)                                                  00190000
      NODE(3)=ITEMP(4)                                                  00200000
      MCODE=ITEMP(5)                                                    00210000
      AKAREA=TEMP(6)                                                    00220000
      BIYY=TEMP(7)                                                      00230000
      AYY=TEMP(8)                                                       00240000
      BIZZ=TEMP(9)                                                      00250000
      AZZ=TEMP(10)                                                      00260000
      BJ=TEMP(11)                                                       00270000
      NFIX1=ITEMP(12)                                                   00280000
      NFIX2=ITEMP(13)                                                   00290000
      NFIX3=ITEMP(14)                                                   00300000
      NFIX4=ITEMP(15)                                                   00310000
      NFIX5=ITEMP(16)                                                   00320000
      NFIX6=ITEMP(17)                                                   00330000
      E=TEMP(18)                                                        00340000
      G=TEMP(19)                                                        00350000
      X1=TEMP(20)                                                       00360000
      Y1=TEMP(21)                                                       00370000
      Z1=TEMP(22)                                                       00380000
      X2=TEMP(23)                                                       00390000
      Y2=TEMP(24)                                                       00400000
      Z2=TEMP(25)                                                       00410000
      X3=TEMP(26)                                                       00420000
      Y3=TEMP(27)                                                       00430000
      Z3=TEMP(28)                                                       00440000
      XA3=TEMP(29)                                                      00450000
```

```
      YA3=TEMP(30)
      ZA3=TEMP(31)
      CALL UNPAK2(ITEMP(32), NX(1))
      CALL UNPAK2(ITEMP(33), NX(7))
      INDEX(1)=ITEMP(34)
      INDEX(2)=ITEMP(35)
      DO 50 J=1,12
      DO 50 I=1,12
      STIFFK(I,J)=0.
      STRESS(I,J)=0.
      AMBDAT(I,J)=0.
   50 AMBDA(I,J)=0.
      IF (MCODE.EQ.IBLANK) GO TO 70
   69 X3=XA3
      Y3=YA3
      Z3=ZA3
   70 ALENTH=SQRT((X1-X2)**2+(Y1-Y2)**2+(Z1-Z2)**2)
      CALL BEMFRM(STIFFK)
      A=Y2*Z3+Y1*Z2+Z1*Y3-(Y2*Z1+Y3*Z2+Y1*Z3)
      B=X1*Z3+Z2*X3+Z1*X2-(X3*Z1+X1*Z2+X2*Z3)
      C=X1*Y2+Y1*X3+X2*Y3-(X3*Y2+X1*Y3+X2*Y1)
      GG=SQRT(A**2+B**2+C**2)
      AMBDA(1,1)=(X2-X1)/ALENTH
      AMBDA(1,2)=(Y2-Y1)/ALENTH
      AMBDA(1,3)=(Z2-Z1)/ALENTH
      AMBDA(3,1)=A/GG
      AMBDA(3,2)=B/GG
      AMBDA(3,3)=C/GG
      AMBDA(2,1)=AMBDA(3,2)*AMBDA(1,3)-AMBDA(1,2)*AMBDA(3,3)
      AMBDA(2,2)=AMBDA(3,3)*AMBDA(1,1)-AMBDA(1,3)*AMBDA(3,1)
      AMBDA(2,3)=AMBDA(3,1)*AMBDA(1,2)-AMBDA(1,1)*AMBDA(3,2)
      DO 15 JK=4,10,3
      DO 15 J=1,3
      JKPLJ=JK+J-1
      DO 15 I=1,3
      IKPLI=JK+I-1
   15 AMBDA(IKPLI,JKPLJ)=AMBDA(I,J)
      DO 16 J=1,12
      DO 16 I=1,12
   16 AMBDAT(I,J)=AMBDA(J,I)
      VOLUME = ALENTH * AZZ
      CALL MATMU2(STIFFK,AMBDA,STRESS,12,12,12)
      CALL MATMU2(AMBDAT,STRESS,STIFFK,12,12,12)
      TEMP(1) = SQRT(ALENTH/(2.*G*BJ))
      TEMP(2) = SQRT(ALENTH/(6.*E*BIYY))
      TEMP(3) = SQRT(ALENTH/(6.*E*BIZZ))
      TEMP(4) =  1./AXAREA
      TEMP(5) = SQRT(ALENTH/(2.*G*AZZ))
      TEMP(6) = SQRT(ALENTH/(2.*G*AYY))
      JK = 1
      JL = JK
   86 DO 87 I=JK,12,6
      DO 87 J=1,12
      IF(TEMP(JL).EQ.0.) GO TO 87
      STRESS(I,J) = STRESS(I,J) * TEMP(JL)
   87 CONTINUE
      JL = JL + 1
      JK = JL
      IF(JL.LE.6) GO TO 86
      DO 20 J=1,12
      STRESS(4,J)=STRESS(5,J)
      STRESS(5,J)=STRESS(6,J)
      STRESS(6,J)=STRESS(8,J)
      STRESS(7,J)=STRESS(9,J)
   20 STRESS(8,J)=STRESS(10,J)
      CALL DANDS(STIFFK,STRESS,12,8,NX,INDEX,NODE,VOLUME)
      RETURN
      END
```

MEMBER NAME BFEM6B1

```
      SUBROUTINE BEMFRM(STIFFK)
      DIMENSION STIFFK(24,24)
      COMMON/COMSB/ALENTH,E,BIYY,AYY,G,BIZZ,AZZ,NFIX1,NFIX2,NFIX3,
     1NFIX4,NFIX5,NFIX6,BJ,AXAREA
      TERM1=ALENTH/(3.*E*BIYY)
      TERM2=1./(AYY*G*ALENTH)
      TERM3=ALENTH/(3.*E*BIZZ)
      TERM4=1./(AZZ*G*ALENTH)
      TAAY=TERM1+TERM2
      TABY=TERM2-.5*TERM1
      TBAY=TABY
      TBBY=TAAY
      TAAZ=TERM3+TERM4
      TABZ=TERM4-.5*TERM3
      TBAZ=TABZ
      TBBZ=TAAZ
      C11Z = 0.
      C12Z = 0.
      C22Z = 0.
      C11Y = 0.
      C12Y = 0.
      C22Y = 0.
      IF((NFIX2.EQ.1).AND.(NFIX5.EQ.0))GO TO 1
      IF((NFIX2.EQ.0).AND.(NFIX5.EQ.1))GO TO 2
      IF((NFIX2.EQ.1).AND.(NFIX5.EQ.1))GO TO 4
      DENOM = TAAY * TBBY - TABY **2
      C11Y = TBBY/DENOM
      C12Y = -TABY/DENOM
      C22Y = TAAY/DENOM
      GO TO 4
    1 C22Y = 1./TBBY
      GO TO 4
    2 C11Y = 1./TAAY
    4 IF((NFIX3.EQ.1).AND.(NFIX6.EQ.0))GO TO 5
      IF((NFIX3.EQ.0).AND.(NFIX6.EQ.1))GO TO 6
      IF((NFIX3.EQ.1).AND.(NFIX6.EQ.1))GO TO 8
      DENOM = TAAZ * TBBZ - TABZ ** 2
      C11Z = TBBZ / DENOM
      C12Z = -TABZ/ DENOM
      C22Z = TAAZ / DENOM
      GO TO 8
    5 C22Z = 1./TBBZ
      GO TO 8
    6 C11Z = 1./TAAZ
    8 IF((NFIX1.EQ.1).AND.(NFIX4.EQ.1))GO TO 9
      STIFFK(1,1)=BJ*G/ALENTH
      GO TO 10
    9 STIFFK(1,1)=0.
   10 STIFFK(1,7)=-STIFFK(1,1)
      STIFFK(7,1)=-STIFFK(1,1)
      STIFFK(7,7)=STIFFK(1,1)
      STIFFK(4,4)=AXAREA*E/ALENTH
      STIFFK(4,10)=-STIFFK(4,4)
      STIFFK(10,4)=-STIFFK(4,4)
      STIFFK(10,10)=STIFFK(4,4)
      STIFFK(2,2)=C11Y
      STIFFK(8,8) = C22Y
      STIFFK(2,6)=-(C11Y+C12Y)/ALENTH
      STIFFK(6,2)=STIFFK(2,6)
      STIFFK(2,12)=-STIFFK(2,6)
      STIFFK(12,2)=-STIFFK(2,6)
      STIFFK(6,8) = -(C12Y+C22Y)/ALENTH
      STIFFK(8,6) = STIFFK(6,8)
      STIFFK(8,12)=-STIFFK(6,8)
      STIFFK(12,8)=-STIFFK(6,8)
      STIFFK(5,5) = (C11Z+2.*C12Z +C22Z)/(ALENTH**2)
      STIFFK(5,11)=-STIFFK(5,5)
      STIFFK(11,5)=-STIFFK(5,5)
      STIFFK(11,11)=STIFFK(5,5)
      STIFFK(2,8)=C12Y
```

```
      STIFFK(8,2)=C12Y                                                  00790000
      STIFFK(3,7)=C11Z                                                  00800000
      STIFFK(9,9) = C22Z                                                00810000
      STIFFK(3,5)=(C11Z+C12Z)/ALENTH                                    00820000
      STIFFK(5,3)=STIFFK(3,5)                                           00830000
      STIFFK(3,11)=-STIFFK(3,5)                                         00840000
      STIFFK(11,3)=-STIFFK(3,5)                                         00850000
      STIFFK(5,9) = (C12Z + C22Z)/ALENTH                                00860000
      STIFFK(9,5) = STIFFK(5,9)                                         00870000
      STIFFK(11,9)= -STIFFK(5,9)                                        00880000
      STIFFK(9,11)= -STIFFK(5,9)                                        00890000
      STIFFK(6,6) = (C11Y + 2.* C12Y + C22Y)/(ALENTH**2)                00900000
      STIFFK(6,12)=-STIFFK(6,6)                                         00910000
      STIFFK(12,6)=-STIFFK(6,6)                                         00920000
      STIFFK(12,12)=STIFFK(6,6)                                         00930000
      STIFFK(3,9)=C12Z                                                  00940000
      STIFFK(9,3)=C12Z                                                  00950000
      RETURN                                                            00960000
      END                                                               00970000
```

```
MEMBER NAME   CD6R1
      SUBROUTINE CD                                                     00010000
      DIMENSION LT(999),NOS(3),E(3),U(3),HOLD(80),G(3)                  00020000
      DIMENSION MT(999),MAP(300),MDOF(300)
      COMMON/JUNE/ KOP,LAC,LT,NIN,NOUT,NPCH,E,U,HOLD,G,JN,MAP,MDOF,MT,MM00030000
      EQUIVALENCE(ID,NOS(1)),(ND1,NOS(2)),(ND2,NOS(3))                  00040000
  901 FORMAT(I3,2I3,3X,I1,E10.5)                                        00050000
  902 FORMAT('01',2I4,5X,8A1,46X,5A1)                                   00060000
   10 READ(0,901)NOS,MC,AEL                                             00070000
      WRITE(2,901)NOS,MC,AEL                                            00070010
      IF(ID.NE.2)GO TO 30                                               00080000
      IF(MC.EQ.0)MC=1                                                   00090000
      ND1=LT(ND1)                                                       00100000
      ND2=LT(ND2)                                                       00110000
      WRITE(0,100)AEL,E(MC)                                             00130000
  100 FORMAT(2PE11.3,2PE9.1)                                            00140000
      READ(0,101)HOLD                                                   00140010
  101 FORMAT(80A1)                                                      00140020
      WRITE(NPCH,902)ND1,ND2,HOLD(2),HOLD(3),(HOLD(I),I=5,9),HOLD(11),  00150000
     1HOLD(13),HOLD(14),HOLD(16),HOLD(17),HOLD(20)                      00160000
   30 RETURN                                                            00170000
      END                                                               00180000
```

```
MEMBER NAME   COORD61
      SUBROUTINE COORD                                                  00010000
      DIMENSION LT(999),NTH(3),NEC(6),NDL(3),HOLD(80),E(3),U(3),G(3)    00090000
      DIMENSION SIGN1(9),MAP(300)                                       00100000
      DIMENSION MDOF(300),MT(999)                                       00110000
      COMMON/JUNE/ KOP,LAC,LT,NIN,NOUT,NPCH,E,U,HOLD,G,JN,MAP,MDOF,MT,MM00120000
      DATA XSIGN/'-'/                                                   00130000
      DATA PSIGN/'+'/                                                   00140000
      DATA AZERO/ZF0404040/                                             00150000
      EQUIVALENCE(NTH(1),NEC(1)),(NDL(1),NEC(4))                        00160000
      MM=0                                                              00170000
      JN = 0                                                            00180000
      DO 110 J=1,300                                                    00190000
  110 MAP(J) = 0                                                        00200000
  100 READ(NIN,101)NDN,X,Y,Z,NEC,II                                     00210000
      WRITE(2,101)NDN,X,Y,Z,NEC,II                                      00210100
```

```
      IF(X.EQ.0.0) X=0.00001
  101 FORMAT(I3,3F10.4,6I1,32X,I1)
      DO 120 J=1,6
      IF(NEC(J).NE.2) GO TO 120
      JN = JN + 1
      MAP(JN) = NDN
      MDOF(JN) = J
  120 CONTINUE
      MM=MM+1
      LT(NDN)=MM
      MT(MM) = NDN
      DO 200 J=1,6
      IF(NEC(J).GE.1) GO TO 150
      NEC(J)=1
      GO TO 200
  150 NEC(J)=0
  200 CONTINUE
      WRITE(6,220)X,Y,Z
  220 FORMAT(F10.4,2X,F10.4,2X,F10.4,2X)
      READ(8,201)HOLD
  201 FORMAT(80A1)
      DO 350 J=2,26,12
      N=J+3
      DO 250 I=J,N
      IF(HOLD(I).EQ.XSIGN.OR.HOLD(I).EQ.PSIGN)GO TO 300
  250 CONTINUE
      GO TO 350
  300 HOLD(J-1)=HOLD(I)
      HOLD(I)=AZERO
  350 CONTINUE
      WRITE(NPCH,351) MM,(HOLD(I),I=1,5),(HOLD(I),I=7,17),
     1(HOLD(I),I=19,29),(HOLD(I),I=31,36),NDL,NTH
  351 FORMAT(      I4,1X,33A1,6I1)
      IF(II.NE.9)GO TO 100
      WRITE(NPCH,361)
  361 FORMAT(80X)
      RETURN
      END
```

MEMBER NAME BANDS6B1

```
      SUBROUTINE BANDS(STIFF,STRESS,NCOLS,NRISM,NX,ID,NODE,VOLUME)
      COMMON NELREC, MRECS, NREC11, LSTCON, NBAND, MXBAND,
     1       MXLCND, MXNDRS, NBASE, MSTIFF, MSTRES, LIN,
     2       NSFREC,HOGO,NSTRPS,KNTLDS,MUDDY1,MUDDY2
      COMMON/RCNT/JRECS
      COMMON/JULY/IDNO
      EQUIVALENCE(NUMREC,JRECS)
      DIMENSION STIFF(24,24),STRESS(24,24),NX(24),EXTR1(24,24)
     1,SAVE(24,24),EXTR2(24,24),NODE(4),ID(4),NID(4)
      DIMENSION NEWNX(24),STRES2(24,24),NNODE(4),NSTIFF(4),NSTRES(4)
     1,IND(24),IN1(24)
      DO 2 J=1,24
      DO 2 I=1,24
      EXTR1(I,J)=0.
      STRES2(I,J) = 0.
      SAVE(I,J)=0.
    2 EXTR2(I,J)=0.
      LL=0
      DO 99 I=1,NBLOCK
      NSTIFF(I)=0
      LL=LL+6
      L=LL-5
      DO 99 M=L,LL
      IF(NX(M).EQ.0)NSTIFF(I)=NSTIFF(I)+1
   99 CONTINUE
      NGRLAT=NODE(1)
      DO 10 I=2,NBLOCK
```

```
      IF(NODE(I).GT.NGREAT)NGREAT=NODE(I)                              00360000
   10 CONTINUE                                                         00370000
      NLEAST=NODE(1)                                                   00430000
      DO 11 I=2,NBLOCK                                                 00440000
      IF(NODE(I).LT.NLEAST)NLEAST=NODE(I)                              00460000
   11 CONTINUE                                                         00470000
      KK=0                                                             00500000
      DO 12 I=1,NBLOCK                                                 00510000
      IF(NODE(I).NE.NGREAT)GO TO 18                                    00530000
      NNODE(1)=NODE(I)                                                 00540000
      NID(1)=ID(I)                                                     00550000
      DO 14 K=1,6                                                      00560000
      NSTRES(1)=NSTIFF(I)                                              00570000
      KKK=K+KK                                                         00580000
      NEWNX(K)=NX(KKK)                                                 00590000
      DO 14 J=1,NRISM                                                  00610000
   14 STRES2(J,K)=STRESS(J,KKK)                                        00620000
      GO TO 20                                                         00630000
   18 KK=KK+6                                                          00640000
   12 CONTINUE                                                         00670000
   20 LL=0                                                             00710000
      DO 30 I=1,NBLOCK                                                 00720000
      IF(NODE(I).NE.NLEAST)GO TO 29                                    00740000
      NNODE(2)=NODE(I)                                                 00750000
      NID(2)=ID(I)                                                     00760000
      NSTRES(2)=NSTIFF(I)                                              00770000
      DO 24 L=1,6                                                      00790000
      LLL=L+LL                                                         00800000
      NEWNX(L+6)=NX(LLL)                                               00810000
      DO 24 J=1,NRISM                                                  00830000
   24 STRES2(J,L+6)=STRESS(J,LLL)                                      00840000
      GO TO 31                                                         00850000
   29 LL=LL+6                                                          00870000
   30 CONTINUE                                                         00890000
   31 IF(NBLOCK.EQ.2)GO TO 51                                          00920000
      MM=0                                                             00940000
      DO 40 I=1,NBLOCK                                                 00960000
      IF(NODE(I).EQ.NGREAT)GO TO 39                                    00970000
      IF(NODE(I).EQ.NLEAST)GO TO 39                                    00980000
      NNODE(3)=NODE(I)                                                 01000000
      NID(3)=ID(I)                                                     01010000
      NSTRES(3)=NSTIFF(I)                                              01020000
      DO 34 M=1,6                                                      01040000
      MMM=M+MM                                                         01050000
      NEWNX(M+12)=NX(MMM)                                              01060000
      DO 34 J=1,NRISM                                                  01080000
   34 STRES2(J,M+12)=STRESS(J,MMM)                                     01090000
      GO TO 41                                                         01100000
   39 MM=MM+6                                                          01110000
   40 CONTINUE                                                         01120000
   41 IF(NBLOCK.EQ.3)GO TO 51                                          01130000
      NN=0                                                             01150000
      DO 50 I=1,NBLOCK                                                 01170000
      IF(NODE(I).EQ.NGREAT)GO TO 49                                    01180000
      IF(NODE(I).EQ.NLEAST)GO TO 49                                    01190000
      IF(NODE(I).EQ.NNODE(3))GO TO 49                                  01200000
      NNODE(4)=NODE(I)                                                 01220000
      NID(4)=ID(I)                                                     01230000
      NSTRES(4)=NSTIFF(I)                                              01240000
      DO 44 N=1,6                                                      01260000
      NNN=N+NN                                                         01270000
      NEWNX(N+18)=NX(NNN)                                              01280000
      DO 44 J=1,NRISM                                                  01290000
   44 STRES2(J,N+18)=STRESS(J,NNN)                                     01300000
      GO TO 51                                                         01310000
   49 NN=NN+6                                                          01320000
   50 CONTINUE                                                         01330000
   51 CONTINUE                                                         01340000
      IND(1) = 1                                                       01360000
      INC = 1                                                          01370000
```

```
      INK =1                                                           01370100
      IN1(1) =1                                                        01370200
      DO 750 I1 = 2,NCOLS                                              01380000
      I1M1= I1-1                                                       01390000
      IF(NX(I1M1).EQ.1) INC=0                                          01400000
      IND(I1) = IND(I1M1) +INC                                         01410000
      INC = 1                                                          01420000
      IF(NEWNX(I1M1).EQ.1) INK =0                                      01420100
      IN1(I1) = IN1(I1M1) +INK                                         01420200
      INK =1                                                           01420300
  750 CONTINUE                                                         01430000
      DO 770  I1=1,NCOLS                                               01440000
      DO 770  J1=1,NCOLS                                               01450000
      EXTR2(IND(I1),IND(J1))=  STIFF(I1,J1)                            01460000
  770 STRES2(I1,IN1(J1)) = STRES2(I1,J1)                               01470000
      INDLST = IND(NCOLS)                                              01470010
      IN1LST = IN1(NCOLS)                                              01470020
      IF(NX(NCOLS).EQ.0) INDLST = INDLST+1                             01470021
      IF(NEWNX(NCOLS).EQ.0)IN1LST = IN1LST+1                           01470025
      IF(INDLST.GT.NCOLS) GO TO 778                                    01470029
      DO 775 I1 = INDLST,NCOLS                                         01470030
      DO 775 J1     =  1 ,NCOLS                                        01470040
      EXTR2(J1,I1) =0.                                                 01470041
  775 EXTR2(I1,J1)= 0.                                                 01470050
  778 IF(IN1LST.GT.NCOLS) GO TO 782                                    01470055
      DO 780  J1=IN1LST,NCOLS                                          01470060
      DO 780  I1=1,NRISM                                               01470070
  780 STRES2(I1,J1) = 0.                                               01470090
  782 CONTINUE                                                         01470095
      JA =0                                                            01470100
      DO 211 I=1,NCOLS                                                 01620000
      IF(NEWNX(I).GT.0) GO TO 211                                      01630000
      JA=JA+1                                                          01640000
      EXTR1(I,JA)=1.0                                                  01650000
  211 CONTINUE                                                         01660000
      WRITE( 9) NUMREC, NBLOCK, NRISM, (NID(I),I=1,NBLOCK),            01690000
     1(NDOF(I),I=1,NBLOCK), (NSTRES(I),I=1,NBLOCK), JA,                01700000
     2((STRES2(I,J),J=1,JA),I=1,NRISM),IDND,VOLUME                     01710000
       MRECS=MRECS+1                                                   01720000
      MROWL=0                                                          01750000
      DO 301 I=1,NBLOCK                                                01760000
      MROWL=MROWL+NSTIFF(I)                                            01770000
      MROWF=MROWL-NSTIFF(I)+1                                          01780000
      MCOLL=0                                                          01790000
      DO 301 J=1,NBLOCK                                                01800000
      IF(NSTIFF(I).EQ.0.OR.NSTIFF(J).EQ.0) GO TO 301                   01810000
      MCOLL=MCOLL+NSTIFF(J)                                            01820000
      MCOLF=MCOLL-NSTIFF(J)+1                                          01830000
      WRITE(8) ID(I),ID(J),NSTIFF(I),NSTIFF(J),((EXTR2(M,N),N=MCOLF,   01860000
     1MCOLL),M=MROWF,MROWL)                                            01870000
      NSFREC = NSFREC + 1                                              01880000
  301 CONTINUE                                                         01900000
      RETURN                                                           01910000
      END                                                              01920000
MEMBER NAME  DATA601
      SUBROUTINE DATA                                                  00010000
      DIMENSION            LT(999),NCG(10),E(3),U(3),HOLD(80),G(3)
      DIMENSION MAP(300),WT(250),WMASS(6),C(250),PH(250,10),
     *              MDOF(300),FOR(250,10),OP(6),MT(999)
      DIMENSION DATA(26)                                               00090100
      COMMON/JUNE/ KOP,LAC,LT,NIN,NOUT,NPCH,E,U,HOLD,G,JN,MAP,MDOF,MT,MM00110000
      COMMON/PART/TITLE(20)                                            00120000
      NPCH = 5                                                         00130000
      NIN = 15                                                         00140000
      CALL LOADER(DATA,1)                                              00150000
      NOSC=DATA(1)                                                     00160000
      NIR=DATA(3)                                                      00170000
      NTC=DATA(14)                                                     00180000
      IPT=DATA(17)                                                     00190000
      NSTIFF=DATA(18)                                                  00200000
```

```
      MSTRES=DATA(26)                                                   00210000
      WRITE(NPCH,1002) (TITLE(I),I=1,20)                                00220000
 1002 FORMAT(20A4)                                                      00230000
      WRITE(NPCH,1003) MSTIFF,MSTRES                                    00240000
 1003 FORMAT( 2I4)                                                      00250000
      WRITE(NPCH,1004) NJSD,NIC                                         00260000
 1004 FORMAT(     '0001',I4,'0000',I4,'00000000')                       00270000
      DO 1050 I=1,3                                                     00280000
      F(I)=DATA(I+6)                                                    00290000
      U(I)=DATA(I+9)                                                    00300000
      G(I)=F(I)/(2.0*(1.0+U(I)))                                        00310000
 1050 CONTINUE                                                          00320000
      CALL COORD                                                        00390000
      WRITE(14) MM, (MT(I),I=1,MM)                                      00400000
 1100 READ(NIN,21720) HOLD                                              00450000
      READ(0,1120)NT,JJ                                                 00460000
 1120 FORMAT(I3,68X,I1)                                                 00470000
      NT=NT+1                                                           00510000
      GO TO(1400,1200,1300,1400,1500),NT                                00530000
 1200 CONTINUE                                                          00540000
      WRITE(   6,1270)                                                  00550000
 1270 FORMAT(1H0 'ERRONEOUS CARD IN INPUT DATA')                        00560000
      CALL EXIT                                                         00570000
 1300 CALL CD                                                           00580000
      GO TO 1610                                                        00590000
 1400 CALL DF                                                           00600000
      GO TO 1610                                                        00610000
 1500 CALL FF                                                           00620000
 1610 IF(JJ.NE.9)GO TO 1100                                             00630000
      WRITE(NPCH,10001)                                                 00640000
10001 FORMAT(80X)                                                       00650000
      DO 11010 J=1,250
11010 WT(J) = 0.0                                                       00710000
11000 READ(NIN,11001) NODE,(WMASS(I),I=1,6), NC                         00720000
11001 FORMAT(I4,6E10.0,15X,I1)                                          00730000
      NN = 0                                                            00740000
      DO 11200 L=1,6                                                    00750000
      IF(WMASS(L).EQ.0.) GO TO 11300                                    00760000
      IF(NN.GT.0) GO TO 11150                                           00770000
      NN = 1                                                            00780000
      JJ = 0                                                            00790000
11100 JJ = JJ + 1                                                       00800000
      IF(MAP(JJ).NE.NODE) GO TO 11100                                   00810000
      GO TO 11200                                                       00820000
11150 JJ = JJ + 1                                                       00830000
11200 WT(JJ) = WMASS(L)                                                 00840000
11300 IF(NC.NE.9) GO TO 11000                                           00850000
      IF(IPT.EQ.1)GO TO 20000                                           01010000
      READ(12,19000)MSO,LRA                                             01020000
19000 FORMAT(2I4)                                                       01030000
      IF(NTR.LE.MSO)GO TO 19100                                         01040000
      WRITE(6,20010)MSO,NTR                                             01050000
      NTR=MSO                                                           01060000
19100 IF(NTC.LE.LRA)GO TO 19200                                         01070000
      WRITE(6,20060)LRA,NTC                                             01080000
      NTC=LRA                                                           01090000
19200 READ(12,19300)NR,NC,(C(ID),ID=1,5)                                01100000
19300 FORMAT(I3,I2,5E15.0)                                              01110000
      IF(NR.LE.0)GO TO 20300                                            01120000
      DO 19400 ID=1,5                                                   01130000
      PH(NR,NC)=C(ID)                                                   01140000
19400 NC=NC+1                                                           01150000
      GO TO 19200                                                       01160000
20000 REWIND 3                                                          01170000
      READ(3) MSO,LRA                                                   01180000
      IF(NTR.LE.MSO)GO TO 20050                                         01190000
      WRITE(6,20010)MSO,NTR                                             01200000
20010 FORMAT(1H0, 4HMSO=,I6,20X, 4HNTR=,I6)                             01210000
      NTR=MSO                                                           01220000
20050 IF(NTC.LE.LRA)GO TO 20100                                         01230000
      WRITE(6,20060)LRA,NTC                                             01240000
20060 FORMAT(1H0, 4HLRA=,I6,20X, 4HNTC=,I6)                             01250000
      NTC = LRA                                                         01260000
```

```
20100 DO 20150 J=1,NTC                                                      01270000
      READ(3) (C(NN),NN=1,NTR)                                              01280000
      DO 20150 NN=1,NTR                                                     01290000
20150 PH(NN,J) = C(NN)                                                      01300000
20300 DO 21000  J=1,NTR                                                     01330000
      DO 21000  K=1,NTC                                                     01340000
      FOR(J,K)=WT(J)*PH(J,K)*10000.                                         01350000
21000 CONTINUE                                                              01350400
      DO 21900 I=1,NTC                                                      01410000
      KP = 1                                                                01450000
      DO 21800 JJ=1,NTR                                                     01460000
      NO = MAP(JJ)                                                          01470000
      DO 21500 K=KP,6                                                       01520000
      IF(K.EQ.MDOF(JJ)) GO TO 21550                                         01530000
      OP(K) = 0.                                                            01540000
21500 CONTINUE                                                              01550000
      GO TO 21600                                                           01560000
21550 OP(K) = FOR(JJ,I)                                                     01570000
21600 IF(MAP(JJ+1).EQ.NO) GO TO 21750                                       01580000
      KO = K+1                                                              01590000
      DO 21650 KK=KO,6                                                      01600000
      OP(KK) = 0.                                                           01610000
21650 CONTINUE                                                              01620000
      KP = 1                                                                01630000
      WRITE(0,21700)LT(NO),I,OP(4),OP(5),OP(6),OP(1),OP(2),OP(3)            01680000
21700 FORMAT(2I4,6PGE12.0)                                                  01690000
      READ(0,21720)HOLD                                                     01690100
      WRITE(NPCH, 21720) HOLD                                               01700000
21720 FORMAT(      80A1)                                                    01710000
      GO TO 21800                                                           01720000
21750 KP = K + 1                                                            01730000
21800 CONTINUE                                                              01740000
21900 CONTINUE                                                              01750000
      WRITE(NPCH,21901)                                                     01760000
21901 FORMAT(80X)                                                           01770000
      REWIND 5                                                              01780000
      RETURN                                                                01790000
      END                                                                   01800000
```

MEMBER NAME DE6A1

```
      SUBROUTINE DE                                                         00010000
      DIMENSION LT(999),MGR(3),NOS(4),TPC(3),HOLD(80),E(3),U(3),G(3)        00020000
      DIMENSION MT(995),MAP(300),MDOF(360)
      COMMON/JUNE/ KOP,LAC,LT,NIN,NOUT,NPCH,E,U,HOLD,G,JN,MAP,MDOF,MT,MM00030000
      EQUIVALENCE (ID,NOS(1)),(ND1,NOS(2)),(ND2,NOS(3)),(ND3,NOS(4)),       00040000
     1(MGR(1),NOS(2))                                                       00050000
  901 FORMAT(I3,3I5,I1,3F10.5)                                              00060000
  902 FORMAT('04',3I4,4X,31A1)                                              00070000
   10 READ(0,901)NOS,MC,TPC                                                 00080000
      WRITE(2,901)NOS,MC,TPC                                                00080010
      IF(ID.NE.0.AND.ID.NE.3)GO TO 30                                       00090000
      IF(MC.EQ.0)MC=1                                                       00100000
      IF(TPC(2).NE.0.0)GO TO 21                                             00110000
      IF(TPC(3).NE.0.0)GO TO 21                                             00120000
      TPC(2)=TPC(1)                                                         00130000
      TPC(3)=TPC(1)                                                         00140000
   21 ND1=LT(ND1)                                                           00150000
      ND2=LT(ND2)                                                           00160000
      ND3=LT(ND3)                                                           00170000
      NBLOCK=NCOLS/6                                                        00180000
      WRITE(0,100)TPC,E(MC),G(MC)                                           00190000
  100 FORMAT(1PE10.3,1PE10.3,1PE10.3,2PE9.1,2PE9.1)                         00200000
      READ(0,101)HOLD                                                       00200010
  101 FORMAT(80A1)                                                          00200020
      WRITE(NPCH,902)MGR,HOLD(2),(HOLD(I),I=4,8),HOLD(10),HOLD(12),         00210000
     1(HOLD(I),I=14,18),HOLD(20),HOLD(22),(HOLD(I),I=24,28),HOLD(30),       00220000
     2HOLD(32),HOLD(33),HOLD(35),HOLD(36),HOLD(39),HOLD(41),HOLD(42),       00230000
     3HOLD(44),HOLD(45),HOLD(48)                                            00240000
   30 RETURN                                                                00250000
      END                                                                   00260000
```

```
MEMBER NAME  EE681
      SUBROUTINE EE                                                     00010000
      DIMENSION BEL(3),LT(999),NOS(4),E(3),U(3),HOLD(80),G(3)           00020000
      DIMENSION MT(999),MAP(300),MUDF(300)
      COMMON/JUDD/ KOP,LAC,LT,NIN,NOUT,NPCH, ,U,HOLD,G,JN,MAP,MUDF,MT,MM00030000
      EQUIVALENCE(ID,NOS(1)),(ND1,NOS(2)),(ND2,NOS(3)),                 00040000
     1(ND3,NOS(4)),(SZZ,BEL(1)),(P,BEL(2)),(AZZ,BEL(3))                 00050000
  901 FORMAT(I3,3I5,I1,3F10.5)                                          00060000
  903 FORMAT('02',3I4,25X,24A1,I6,10A1)                                 00070000
   10 READ(G,901)NOS,MC,BEL                                             00080000
      WRITE(2,901)NOS,MC,BEL                                            00080010
      NUMBER=610010                                                     00090000
      IF(ID.NE.4)GO TO 30                                               00100000
      IF(MC.EQ.0)MC=1                                                   00110000
      IF(P.EQ.0)NUMBER=110110                                           00120000
      ND1=LT(ND1)                                                       00130000
      ND2=LT(ND2)                                                       00140000
      ND3=LT(ND3)                                                       00150000
      WRITE(0,100)SZZ,AZZ,P,E(MC),G(MC)                                 00170000
  100 FORMAT(2PF11.3,2PF11.3,2PF11.3,2PF9.1,2PF9.1)                     00180000
      READ(0,101)HOLD                                                   00180010
  101 FORMAT(80A1)                                                      00180020
      WRITE(NPCH,903)ND1,ND2,ND3,HOLD(2),HOLD(3),(HOLD(I),I=5,9),       00190000
     1HOLD(11),HOLD(13),HOLD(14),(HOLD(I),I=16,20),HOLD(22),            00200000
     2HOLD(24),HOLD(25),(HOLD(I),I=27,31),HOLD(33),NUMBER,HOLD(35),     00210000
     3HOLD(36),HOLD(38),HOLD(39),HOLD(42),HOLD(44),                     00220000
     4HOLD(45),HOLD(47),HOLD(48),HOLD(51)                               00230000
   30 RETURN                                                            00240000
      END                                                               00250000


MEMBER NAME  GETNGR1
      SUBROUTINE GENRTN                                                 00010000
      COMMON NELREC, NELCS, NPEC11, LSTCON, KHAND, MXBAND,              00040000
     1       MXLCND, MYNODS, NBASE, MSTIFF, NSTRES, LIN,                00050000
     2       LSFPEC,NCGO,NSTRPS,KNTLDS,MUDDY1,MUDDY2                    00060000
      COMMON /BLOCK5/  ITEMP(68)                                        00070000
      COMMON/BLOCK2/        NX(24),S11(24,24),S12(24,24),ALLDA1(24,24)00080000
     1,ALFDA(24,24),AK21(24,24),AK12(24,24),AKINV(24,24),AK(24,24),     00090000
     2DMAT(24,24),NODE(4)                                               00100000
      COMMON/PCNT/JRECS                                                 00110000
      COMMON/JULY/IDNO                                                  00120000
      DIMENSION TEMP(68)                                                00130000
      EQUIVALENCE(TEMP(1),ITEMP(1))                                     00140000
      NSEFEC = 0                                                        00150000
      MRECS=0                                                           00160000
      JRECS=0                                                           00170000
      REWIND 4                                                          00180000
      REWIND 11                                                         00190000
    1 IF(JRECS.EQ.NELREC) GO TO 11                                      00200000
      READ (4) (ITEMP(I),I=1,68)                                        00210000
       JRECS=JRECS+1                                                    00220000
      IDNO=ITEMP(1)                                                     00230000
      GO TO (2,3,4,5,6,7,8,9,10),IDNO                                   00240000
    2 CALL AXIAL                                                        00250000
      GO TO 1                                                           00260000
    3 CALL BEAM                                                         00270000
      GO TO 1                                                           00280000
    4 CONTINUE                                                          00290000
    5 CALL TRIANG                                                       00310000
      GO TO 1                                                           00320000
    6 CONTINUE                                                          00330000
    7 CONTINUE                                                          00442100
    8 CONTINUE                                                          00442200
    9 CONTINUE                                                          00442300
   10 CONTINUE                                                          00442400
   11 CONTINUE                                                          00450000
      REWIND 4                                                          00460000
      LARR = 9700                                                       00470000
      NZERO = NSTRPS                                                    00480000
      WRITE( 9) NZERO,NZERO, NZERO, LARR, NZERO,                        00490000
```

```
     1        NELREC, NRECS, NREC11, LSTCON, NBAND, MXBAND,               00500000
     1        MXLCND, MXNODS, NBASE, MSTIFF, MSTRES, LIN,                 00510000
     2        NSEREC, NOGO, NSTRPS, KNILDS, MUDDY1, MUDDY2                00520000
      REWIND 8                                                            00530000
      REWIND 9                                                            00540000
      CALL EXIT                                                           00550000
      RETURN                                                              00560000
      END                                                                 00570000

MEMBER NAME  HORWZERO
         TITLE 'SUBROUTINE HORWZERO  **  B C S   P R O P R I E T A R Y  ***' UN022575
* * * * * * * * * * * * * * * * * * * * * * * * * * *                   DN100374
*                                                   *                   DN100374
*          THIS SUBROUTINE SOURCE LISTING IS        *                   DN100374
*          - B C S   P R O P R I E T A R Y -        *                   DN100374
*                                                   *                   DN100374
* * * * * * * * * * * * * * * * * * * * * * * * * * *                   DN100374
*                                                                       DN100374
*              BCS READ/WRITE LOGICAL FORTRAN UNIT ZERO
*              FOR INTERNAL CONVERSION BY FORMAT CODES
*              VERSION 2 LEVEL RELEASE 21.6, AUGUST 1972
*              REVISED BY NORMAN MENZIE       WICHITA DISTRICT
*              THIS ASSUMES THAT THE EXTENDED ERROR FACILITY IS SYSGEN
*              OPTION
*
*              BUFFERS        VALID READ (VR)      READ NON ZERO UNIT
*                   /CONVRT/ INTERNAL CONVERSION BUFFER (ICB)
*
HORWZERO START
         ENTRY FIOCS#                                                   00001100
         EXTRN HORZIOCS
         ENTRY ZROBFR                                                   00001300
         ENTRY HORWFN
         ENTRY HORWFR
BUFPTRS  EQU   X'114'     LOCATION IN IBCOM CONTAINING BUFFER , LENGTH
DSRNLOC  EQU   X'3C'         SAVED DSRN UNIT IN IBCOM FOR MESSAGES
SAVELOC  EQU   X'7C'        SAVE LOCATION IN IBCOM TH ALLOW NEXT I/O
INIT     EQU   X'00'                                                    00001500
         USING *,1                                                      00001600
FIOCS#   EQU   *                                                        00001700
ZZZ      STM   4,15,SAVE                                                00001800
         PROP  1                                                        00001900
         LR    11,1                                                     00002000
         USING ZZZ,11                                                   00002100
         L     1,ADFIOCS                                                00002300
         CLI   DISABLE,X'FF'
         BE    XIT             GO DIRECT TO FIOSH WITHOUT FURTHER CHECKS
         USING CONVRT,12
         L     12,CONVRTAD     SET UP CONVRT BUFFER
         LR    4,0                                                      00002200
         LR    5,2                                                      00002400
         CLI   0(4),X'03'            CHECK FOR A CONTROL I/O STATEMENT  00002500
         BNL   SETOFF         YES---CANCEL BUFFERS IF UNIT IS SAME AS VR 00002600
         CLI   0(4),INIT      NO     CHECK FOR INITIALIZATION CALL      00002700
         BNE   WORK                  NO                                 00002800
         SR    6,6
         ST    6,KODE          ASSUME NO ERRORS
         LA    7,CHECKZ        SET FOR DSRN CHECK
         TM    0(5),X'0F'     YES    CHECK FOR READ-PUNCH TYPE          00002900
         BZ    DSRN                  NO                                 00003000
         NI    199,0                 YES                                00003100
         B     XIT                                                      00003200
SETOFF   EQU   *                                                        00004000
         CLI   0(4),X'04'           IS IT A CLOSE ALL DATA SETS
         BE    CLOSEA             YES
         BAL   7,DSRN
         CLI   3(5),0               IS IT ZERO DSRN
         BE    SETERR3            YES
```

```
MEMBER NAME  HCRUZERO
         CLC   2(2,5),DSRNVR      IS IT SAME UNIT AS FOR -VR-
         BNE   XIT                NO
CLOSEA   EQU   *
         LH    6,K132             RESET LENGTH
         ST    6,SV2                    AND
         L     6,CONVETAD               BUFFER
         ST    6,SV1
         XC    199,199
         B     XIT
SETERR3  EQU   *
         LA    6,3                INDICATE CONTROL OP
SETERR   EQU   *
         ST    6,KODE             ON DSRN (0)
         STC   6,KODEFT                 STORE CODE INTO FORMAT
         LA    6,1
         AH    6,ERRCOUNT
         STH   6,ERRCOUNT               CURRENT COUNT OF ERRORS
         CH    6,ERRLIMIT               HAVE WE REACHED A MAXIMUM
         BNH   LEAVE              NO
         STAE  0                  NULLIFY FORTRAN STAE
         OI    KODEFT,X'F0'             ADD ZONE TO IT
         L     6,VUATEL
         LH    7,0(6)                   GET THE ERROR UNIT
         STH   7,ERRUNIT+2
         L     15,VIOCOM
         MVI   SAVELOC(15),X'FF'  SAVE LOC IN IOCOM--TO ALLOW NEXT I/O
         CNOP  0,4
         BAL   14,4(15)                 GO TO IOCOM WITH FORMAT CALL
ERRUNIT  DC    F'0'                     LOGICAL UNIT
         DC    A(FORMAT)                  FORMAT LOCATION
         BAL   14,16(15)                END THE LIST FOR THIS OUTPUT
         ABEND 993,DUMP
VIOCOM   DC    V(IOCOMM)
VUATEL   DC    V(IHCUATBL)
FORMAT   DC    X'02182RFD'
         DC    C'INCORRECT USE OF READ/WRITE ZERO   CODE= '
KODEFT   DC    X'0022'            END OF THE FORMAT
DSRN     TM    2(5),X'01'          IS IT VARIABLE                  00003400
         BZ    NOTVAR              NO                              00003500
         L     5,0(5)                                              00003600
NOTVAR   EQU   *                                                   00003700
         BR    7
CHECKZ   EQU   *
         CLI   3(5),0       IS IT UNIT ZERO
         BE    BYPASS1             YES                             00003900
         XC    199(2),199          ZERO   TWO FLAG BYTES
         CLI   1(4),X'F0'         IS IT FORMATTED INPUT            00004200
         BNE   WORK               NO                               00004300
         MVI   199+1,X'FF'        SET INPUT FLAG                   00004400
WORK     EQU   *                                                   00004500
         TM    199,X'FE'          WAS IT A ZERO DSRN               00004600
         BO    BYPASS              YES                             00004700
         MVC   ARG,0(4)                                            00004800
         BALR  0,1                GO TO FIOSU                      00004900
ARG      DS    CL2                                                 00005000
         B     LEAVE2             EXTENDED ERROR IN REL 15/16      00005100
         CLC   ARG,EXTREAD              NORMAL RETURN              00005200
         BE    SAVPTR           SAVE BUFFER POINTER AND LENGTH     00005300
         CLI   199+1,X'FF'                                         00005400
         BNE   LEAVE                                               00005500
SAVPTR   EQU   *                2 CONTAINS THE BUFFER LOCATION     00005600
         STM   2,3,SV1          3 CONTAINS THE BUFFER LENGTH       00005700
         MVI   199+2,X'FF'      INDICATE VR BUFFER
         MVI   199+3,X'FF'        INDICATE THERE IS A BUFFER AVAILABLE
         MVC   DSRNVR+2(5)        SAVE THE LAST -VR- UNIT
         B     LEAVE                                               00005800
               VALID READ OVERRIDES ANY OTHER BUFFER POINTER
LEAVE2   LA    1,2(4)             ERROR RETURN   AT CALLS IN IOCOM 00005900
         B     XIT                RESTORE AND RETURN               00006000
BYPASS1  CLI   1(4),X'F1'           IS IT A WRITE ZERO
         BNE   BYPASS           NO
         CLI   199+2,X'0F'            DO WE HAVE A ZROBFR TO USE
         BE    BYPASS2            YES
```

```
         CLC   SV2+2(2),K132     IS VR LENGTH LONGER THAN ICB
         BH    BYPASS         YES
         LH    6,K132              GET ICB LENGTH
         ST    6,SV2               RESET THE POINTERS
         MVC   SV1,CONVRTAD
         XC    DSRNVR,DSRNVR       ZERO OUT THE LAST UNIT NUMBER
         MVI   199+2,0
BYPASS2  MVI   199+3,X'FF'         INDICATE THERE IS A BUFFER AVAILABLE
BYPASS   EQU   *              BYPASS FIOCS                          00006100
         L     6,DSRNPTR     LOAD ADDRESS IN IBCOM TO INDICATE '0' DSRN
         XC    DSRNLOC(4,6),DSRNLOC(6)   SET IT FOR AN IHC212 MESSAGE
         LM    2,3,SV1       BUFFER AND LENGTH--WHY LOAD IT EVERYTIME 00006200
         STM   2,3,BUFPTRS(6)   .   SAVE THESE IN IBCOM FOR ERRMON
         OI    199,X'FF'                                            .00006300
         TM    199+3,X'FF'         DO WE HAVE A BUFFER AVAILABLE
         BNO   SETERR1             NO
LEAVE    LA    1,6(4)              NORMAL RETURN   TO IBCOM         00006400
XIT      LM    4,15,SAVE                                            00006500
         BCR   15,1    THIS ASSUMES REG '0' IS MAINTAINED THRU PROGRAM
*                       FOR FIOCS'S RETURN ADDRESS
SETERR1  LA    6,1            INDICATE THERE IS NO BUFFER FOR READ '0'
         B     SETERR
*ZROBFR  EQU   *              DEFINE USER BUFFER WITH LENGTH
         USING ZROBFR,15                                            00006800
ZROBFR   SAVE  (14,12),,*
         LM    4,5,0(1)                                             00007000
         ST    4,SV1                                                00007100
         L     5,0(5)                                               00007200
         ST    5,SV2                                                00007300
         XC    DSRNVR,DSRNVR       ZERO OUT THE LAST UNIT NUMBER
         MVI   199+3,X'FF'         INDICATE THERE IS A BUFFER AVAILABLE
         MVI   199+2,X'OF'    INDICATE BUFFER USED
         RETURN (14,12)                                             00007400
*HORWEN  EQU   *              TURN      OFF/ON  ALL ENTRIES TO R-W ZERO
         USING HORWEN,15
HORWEN   SAVE  (0,2),,*
         L     2,0(1)         GET PARM  'I'
         L     2,0(2)
         NI    DISABLE,X'00'  I.NE.0     ENABLE  R-W ZERO
         LTR   2,2            WAS I=0
         BZ    HORWEN1        NO
         OI    DISABLE,X'FF'  I .EQ. 0  DISABLE R-W ZERO
HORWEN1  RETURN (0,2)
         USING HORWER,15
HORWER   SAVE  (0,2),,*
         L     2,0(1)              GET THE NEW ERROR LIMIT
         L     2,0(2)
         STH   2,ERRLIMIT          LIMIT OF ERRORS BEFOR ABEND
         RETURN (0,2),1
SAVE     DC    12F
DSRNPTR  EQU   VIBCOM
ADFIOCS  DC    A(HORZIOCS)
199      DC    F'0'                                                 00007700
*                     BYTE 1    'FF' SET AT BYPASS IF DSRN IS ZERO
*                          2    FF   INPUT (FORMATTED)
*                          3   '00'   ICB BUFFER   'FF'  VR BUFFER
*                          3   '0F'   ZROBFR
*                          4   '00'  NO BUFFER SET FOR READ '0'
*                          4.  'FF'  BUFFER HAS BEEN SET FOR '0'
SV1      DC    V(CONVRT)            SET DEFAULT BUFFER
SV2      DC    F'132'               SET DEFAULT LENGTH
CONVRTAD DC    V(CONVRT)
FMTREAD  DC    XL2'00F0'                                            00008100
ERRCOUNT DC    H'0'                ERROR COUNT
ERRLIMIT DC    H'5'           ERROR LIMIT BEFOR ABEND -DEFAULT 5
K132     DC    H'132'
DSRNVR   DC    H'0'           BUFFER UNIT ON VR
DISABLE  DC    X'00'          DEFAULT TO  '0' WHICH INDICATES  'ON'
         SPACE 2
CONVRT   CSECT
         DS    33F
KORE     DS    F
         END                                                        00008200
```

```
MEMBER NAME  INPUT6B1
      SUBROUTINE INPUT
      COMMON NELREC, NRECS, NREC11, LSTCON, NBAND, MXBAND,
     1       MXLCND, MXNODS, NBASE, MSTIFF, MSTRES, LIN,
     2       NSFREC,NOGO,NSTRPS,KNTLDS,MUDDY1,MUDDY2
      DIMENSION        X(1500),Y(1500),Z(1500),XA(1500),YA(1500),ZA(1500)
     1  ,DUMMY(20),INDEX(1500),KONEQ(1500)
      DIMENSION ITYPE(3),JTYPE(2)
      DIMENSION TEMP(68) ,ITEMP(68),INFOCN(6)
      DIMENSION KNUM(155)
      DATA  SEGM/'SEGM'/,END/'   E'/
      EQUIVALENCE (TEMP(1),ITEMP(1))
      LOGG = 0
      NCARD=0
      NELREC = 0
      MXBAND = 200
      NBAND  = 0
      IN = 5
      MXLCND = 100
      MXNODS = 1500
      NREC11=0
      NTEMP=0
      KONT = 52
      DO 1111 I = 1, MXNODS
      X(I)=0.
      Y(I)=0.
      Z(I)=0.
      XA(I)=0.
      YA(I)=0.
      ZA(I)=0.
      KK = 0
      K  = 0
      KONEQ(I) = 0
 1111 INDEX(I)=0
      DO 1112 I=1,13
 1112 DUMMY(I)=0.
      INDEX(1)=1
      READ (5,130) (ITEMP(I),I=1,20)
      WRITE(6,132) (ITEMP(I),I = 1,20)
      DATA IBLANK /1H /
      DATA IA /1HA/
      READ(5,131)MSTIFF,MSTRES
  131 FORMAT(2I4)
       READ(5,100)NBASE,NSTNDS,NANODS,LSTCON,MUDDY2,MUDDY1
      WRITE (6,500)
      NSTRPS = NSTNDS
      IF (MUDDY1 .EQ.1) IN =3
      WRITE (6,501)NBASE,NSTNDS,NANODS,LSTCON
      WRITE (6,505)
      WRITE (6,507)
 1103 FORMAT(I4,19A4)
  103 FORMAT(  A1, 3E11.6, I6)
    1 READ(5,1103) NONO,  (DUMMY(I),I=1,19)
      IF(DUMMY(1).EQ.END.OR.NONO.EQ.0) GO TO 1203
      WRITE(6,91476)(DUMMY(I),I=1,19)
91476 FORMAT(19A4)
      READ(0,103) ICODE,XDUM,YDUM,ZDUM,KONDUM
      GO TO 7
 1203 WRITE(6,1213) (DUMMY(I),I=1,19)
    5 IF(K.EQ.NANODS)GO TO 6
      WRITE(6,101)K
      NOGO=1
    6 IF(KK.EQ.NSTNDS)GO TO 3
      WRITE(6,102)KK
      NOGO=1
    3 CONTINUE
      DO 5001 I=1,NSTNDS
      DO 5001 J=1,NSTNDS
      IF(J.EQ.I) GO TO 5000
      IF(X(J).NE.X(I)) GO TO 5000
      IF(Y(J).NE.Y(I)) GO TO 5000
      IF(Z(J).NE.Z(I)) GO TO 5000
      NOGO=1
      WRITE(6,5010)J,I
```

```
5010 FORMAT(/25X,4HNODE,I5,29H HAS THE SAME COORDS. AS NODE,I5/)        00770000
5000 IF(X(J).NE.XA(I))GO TO 5001                                        00790000
     IF(Y(J).NE.YA(I))GO TO 5001                                        00800000
     IF(Z(J).NE.ZA(I))GO TO 5001                                        00810000
     NOGO=1                                                             00820000
     WRITE(6,5011)J,I                                                   00830000
5011 FORMAT(/25X,4HNODE,I5,31H HAS THE SAME COORDS. AS A-NODE,I5/)      00840000
5001 CONTINUE                                                           00850000
     DO 5020 K=1,NANODS                                                 00860000
     DO 5020 L=1,NANODS                                                 00870000
     IF(K.EQ.L)GO TO 5020                                               00880000
     IF(XA(K).EQ.0.) GO TO 5020                                         00890000
     IF(XA(K).NE.XA(L)) GO TO 5020                                      00900000
     IF(YA(K).NE.YA(L)) GO TO 5020                                      00910000
     IF(ZA(K).NE.ZA(L)) GO TO 5020                                      00920000
     NOGO=1                                                             00930000
     WRITE(6,5021)K,L                                                   00940000
5021 FORMAT(/25X,6HA-NODE,I5,31H HAS THE SAME COORDS. AS A-NODE,I5/)    00950000
5020 CONTINUE                                                           00960000
     LIN=INDEX(NSTNDS+1)                                                00970000
     IF(NOGO.EQ.1) GO TO 1214                                           00990000
     L47 = 155                                                          01060000
     LARR = 2600                                                        01070000
     LX = NSTNDS + 1                                                    01080000
     I1 = LX                                                            01090000
     J2 = L47                                                           01100000
 705 CONTINUE                                                           01110000
     I2 = I1 - 1                                                        01120000
     I1 = I1 - L47                                                      01130000
     IF(I1.GE.1) GO TO 710                                              01140000
     I1=1                                                               01150000
     J2=I2                                                              01160000
 710 CONTINUE                                                           01170000
     DO 720 L =1,J2                                                     01180000
     LX = LX-1                                                          01190000
     KDUM(L) = KONEQ(LX)                                                01200000
 720 CONTINUE                                                           01210000
     LARR = LARR-1                                                      01220000
     WRITE( 9)NSTNDS,I1,I2,LARR,J2,(KDUM(L),L=1,J2)                     01230000
     IF(I1.NE.1)GO TO 705                                               01240000
     WRITE(6,508)                                                       01250000
     GO TO 1214                                                         01260000
   7 N = NONO - NBASE + 1                                               01280000
     KONT = KONT - 2                                                    01290000
     IF(ICODE.EQ.IBLANK) GO TO 11                                       01300000
  10 K=K+1                                                              01340000
     XA(K) = XDUM                                                       01350000
     YA(K) = YDUM                                                       01360000
     ZA(K) = ZDUM                                                       01370000
     IF(KONT)  730,730,731                                              01370100
 730 WRITE(6,129)                                                       01370200
     WRITE(6,507)                                                       01370300
     KONT = 52                                                          01370400
 731 WRITE(6,506) NODE, ICODE, XA(N), YA(N), ZA(N)                      01380000
     GO TO 1                                                            01390000
  11 KK=KK+1                                                            01420000
     IF((N-NTEMP).EQ.1)GO TO 4                                          01430000
     NOGO=1                                                             01440000
     WRITE(6,9450)                                                      01450000
9450 FORMAT(//42H A NODE NUMBER SEQUENCE BREAK HAS OCCURRED//)          01460000
   4 NTEMP=N                                                            01470000
     X(N) = XDUM                                                        01480000
      ENT     4    22    52   236    15    10    0.
      ENT     4    22    52   236    15    11    0.
     Y(N) = YDUM                                                        01490000
     Z(N) = ZDUM                                                        01500000
     KONEQ(N) = KONDUM                                                  01510000
     CALL UNPAK (KONEQ(N), INEQCN, KEEP)                                01520000
     INDEX(N+1) = INDEX(N) + KEEP                                       01530000
     IF(KONT) 732,732,733                                               01530100
```

```
  732 WRITE(6,129)                                                        01530200
      WRITE(6,507)                                                        01530400
      FONT = 52                                                           01530600
  733 WRITE(6,510) NONO,X(N),Y(N),Z(N),(INEQCN(11),11=1,6),INDEX(N+1)     01540000
      GO TO 1                                                             01550000
 1212 WRITE(6,1213)        (DUMMY(I),I=1,20)                              01570050
 1213 FORMAT(1H1,2X,20A4//)                                               01575000
 1214 K1 = 1                                                              01575100
      K2 = 1                                                              01575200
      K4 = 1                                                              01575300
      K5 = 1                                                              01575400
      K6 = 1                                                              01575500
      K7 = 1                                                              01575600
      K8 = 1                                                              01575700
      LINES = 2                                                           01575800
   12 READ(5,111)IDNO,(DUMMY(I),I=1,20)                                   01580000
      WRITE(6,91477)(DUMMY(I),I=1,20)                                     01580100
91477 FORMAT(19A4,A2)                                                     01580110
      IF(DUMMY(1).EQ.IEGE) GO TO 1212                                     01585000
      IF(IDNO.EQ.0) GO TO 2000                                            01590000
   17 NCARD = NCARD + 1                                                   01600080
      NELREC = NELREC + 1                                                 01610000
C                                                                         01615000
C     DATA IS NOW DIRECTED TO APPROPRIATE PART OF PROGRAM ACCORDING TO    01615001
C     WHICH TYPE OF ELEMENT IT IS.                                        01615002
      GO TO (13,13,13,200,200),IDNO                                       01620000
   13 READ(5,112)NODE1,NODE2,NODE3,MCODE,AXAREA,BIYY,AYY,BIZZ,AZZ,BJ,     02560000
     1NFIX1,NFIX2,NFIX3,NFIX4,NFIX5,NFIX6,E,G                             02570000
      IF(IDNO.EQ.1.) GO TO 4000                                           02572000
      IF(K2.NE.1) GO TO 4002                                              02573000
      WRITE(6,127)                                                        02574000
 4002 WRITE(6,120)NCARD,IDNO,NODE1,NODE2,NODE3,MCODE,AXAREA,BIYY,AYY,     02580000
     1BIZZ,AZZ,BJ,NFIX1,NFIX2,NFIX3,NFIX4,NFIX5,NFIX6,E,G                 02590000
      K2 = K2 + 1                                                         02590500
      GO TO 126                                                           02591000
 4000 IF(K1.NE.1) GO TO 4004                                              02592000
      WRITE(6,123)                                                        02593000
 4004 WRITE(6,125)NCARD,IDNO,NODE1,NODE2,AXAREA,E                         02594000
      K1 = K1 + 1                                                         02597000
  126 IF(LINES.GE.23) LINES = 1                                           02597500
      IF(LINES.NE.1) GO TO 4006                                           02598000
      WRITE(6,4007)                                                       02598500
      GO TO (4010,4020,4020),IDNO                                         02598600
 4010 WRITE (6,123)                                                       02598700
      GO TO 4006                                                          02598800
 4020 WRITE (6,127)                                                       02599000
 4007 FORMAT(1H1)                                                         02599200
 4006 IF(IDNO.EQ.1.OR.IDNO.EQ.2) GO TO 4008                               02599500
      LINES = LINES + 1                                                   02599600
 4008 LINES = LINES + 1                                                   02599700
      NODE1 = NODE1 - NBASE + 1                                           02600000
      NODE2=NODE2-NBASE+1                                                 02610000
      NODE3=NODE3-NBASE+1                                                 02620000
      IF(X(NODE1).EQ.0.)GO TO 8419                                        02630000
      IF(X(NODE2).EQ.0.)GO TO 8419                                        02640000
      IF(NODE1.LE.0)GO TO 8401                                            02650000
      IF(NODE2.LE.0)GO TO 8401                                            02660000
      IF(AXAREA.LT.0.)  GO TO 8404                                        02670000
      IF(BIYY.LT.0.)GO TO 8405                                            02680000
      IF(AYY.LT.0.)GO TO 8406                                             02690000
      IF(BIZZ.LT.0.)GO TO 8407                                            02700000
      IF(AZZ.LT.0.)GO TO 8408                                             02710000
      IF(BJ.LT.0.)GO TO 8409                                              02720000
      IF(E.LT.0.)GO TO 8416                                               02730000
      IF(G.LT.0.)GO TO 8417                                               02740000
      NSMALL = MIN0(NODE1,NODE2)                                          02780000
      NLARGE = MAX0(NODE1,NODE2)                                          02790000
      NB1 = INDEX(NLARGE+1) - INDEX(NSMALL)                               02800000
      IF(KONEQ(NLARGE).EQ.111111) CALL UNPAK(KONEQ(NSMALL),INEQCN,NB1)    02800500
      IF(KONEQ(NSMALL).EQ.111111) CALL UNPAK(KONEQ(NLARGE),INEQCN,NB1)    02800600
```

```
      IF(PP1.LE.NBAND)GO TO 60                                          02810000
      IF(PP1.LE.MXBAND)GO TO 51                                         02820000
      NOGO = 1                                                          02830000
      WRITE(6,115)NCARD                                                 02840000
      GO TO 60                                                          02850000
   51 NBAND = NP1                                                       02860000
   60 CONTINUE                                                          02870000
      IF(IDNO.EQ.1)GO TO 219                                            02900000
      IF(NCODE.EQ.IA) GO TO 67                                          02910000
   66 IF (NCODE.NE.IBLANK) GO TO 8421                                   02920000
   67 IF(NFIX1.NE.NFIX4)GO TO 8422                                      02930000
      IF(EIYY.NE.0.)GO TO 72                                            02940000
      IF(AYY.NE.0.)GO TO 8423                                           02950000
      IF(NFIX2.NE.1)GO TO 8423                                          02960000
      IF(NFIX5.NE.1)GO TO 8423                                          02970000
   72 IF(EIZZ.NE.0.)GO TO 76                                            02980000
      IF(AZZ.NE.0.)GO TO 8424                                           02990000
      IF(NFIX3.NE.1)GO TO 8424                                          03000000
      IF(NFIX6.NE.1)GO TO 8424                                          03010000
   76 IF(EJ.LE.0.)GO TO 79                                              03020000
      IF(NFIX1.NE.1)GO TO 8427                                          03030000
      IF(NFIX4.NE.1)GO TO 8427                                          03040000
   79 CONTINUE                                                          03050000
      IF(IDNO.LE.2)GO TO 218                                            03070000
      IF(XA(NODE1).EQ.0.)GO TO 8419                                     03090000
      IF(XA(NODE2).EQ.0.)GO TO 8419                                     03100000
      IF(NOGO-1)217,12,217                                              03360000
 8401 WRITE(6,9401)                                                     03380000
      NOGO=1                                                            03390000
      GO TO 12                                                          03400000
 8404 WRITE(6,9404)                                                     03410000
      NOGO=1                                                            03420000
      GO TO 12                                                          03430000
 8405 WRITE(6,9405)                                                     03440000
      NOGO=1                                                            03450000
      GO TO 12                                                          03460000
 8406 WRITE(6,9406)                                                     03470000
      NOGO=1                                                            03480000
      GO TO 12                                                          03490000
 8407 WRITE(6,9407)                                                     03500000
      NOGO=1                                                            03510000
      GO TO 12                                                          03520000
 8408 WRITE(6,9408)                                                     03530000
      NOGO=1                                                            03540000
      GO TO 12                                                          03550000
 8409 WRITE(6,9409)                                                     03560000
      NOGO=1                                                            03570000
      GO TO 12                                                          03580000
 8416 WRITE(6,9416)                                                     03590000
      NOGO=1                                                            03600000
      GO TO 12                                                          03610000
 8417 WRITE(6,9417)                                                     03620000
      NOGO=1                                                            03630000
      GO TO 12                                                          03640000
 8419 WRITE(6,9419)                                                     03680000
      NOGO=1                                                            03690000
      GO TO 12                                                          03700000
 8421 WRITE(6,9421)                                                     03710000
      NOGO=1                                                            03720000
      GO TO 12                                                          03730000
 8422 WRITE(6,9422)                                                     03740000
      NOGO=1                                                            03750000
      GO TO 12                                                          03760000
 8423 WRITE(6,9423)                                                     03770000
      NOGO=1                                                            03780000
      GO TO 12                                                          03790000
 8424 WRITE(6,9424)                                                     03800000
      NOGO=1                                                            03810000
      GO TO 12                                                          03820000
 8427 WRITE(6,9427)                                                     03830000
```

```
      NOGO=1                                                            03840000
      GO TO 12                                                          03850000
 8438 WRITE(6,9438)                                                     03920000
      NOGO=1                                                            03930000
      GO TO 12                                                          03940000
 8446 WRITE(6,9446)                                                     03950000
      NOGO=1                                                            03960000
      GO TO 12                                                          03970000
  200 READ(5,114)NODE1,NODE2,NODE3,NODE4,TS,TX,TY,E,G                   04020000
      IF(IDNO.EQ.4) GO TO 7001                                          04022000
      IF(K5.NE.1) GO TO 7006                                            04023000
      WRITE(6,7004)                                                     04024000
 7006 WRITE(6,122) NCARD,IDNO,NODE1,NODE2,NODE3,NODE4,TS,TX,TY,E,G      04030000
      K5 = K5 + 1                                                       04030500
      KNODE1 = NODE1 - NBASE + 1                                        04030510
      KNODE2 = NODE2 - NBASE + 1                                        04030520
      KNODE3 = NODE3 - NBASE + 1                                        04030530
      KNODE4 = NODE4 - NBASE + 1                                        04030540
      GO TO 7005                                                        04031000
 7001 IF(K4.NE.1) GO TO 7003                                            04032000
      WRITE(6,7002)                                                     04033000
 7003 WRITE(6,7000) NCARD,IDNO,NODE1,NODE2,NODE3,TS,TX,TY,E,G           04034000
      K4 = K4 + 1                                                       04035000
 7005 IF(LINES.GE.23) LINES = 1                                         04036000
      IF(LINES.NE.1)  GO TO 7008                                        04037000
      WRITE(6,4007)                                                     04038000
      GO TO (7008,7008,7008,7010,7020),IDNO                             04038200
 7010 WRITE (6,7002)                                                    04038400
      GO TO 7008                                                        04038600
 7020 WRITE (6,7004)                                                    04038800
 7008 IF(IDNO.EQ.4) GO TO 7009                                          04039000
      LINES = LINES + 1                                                 04039200
 7009 LINES = LINES + 1                                                 04039400
      NODE1=NODE1-NBASE+1                                               04040000
      NODE2=NODE2-NBASE+1                                               04050000
      NODE3=NODE3-NBASE+1                                               04060000
      IF(NODE1.LE.0)GO TO 8401                                          04070000
      IF(NODE2.LE.0)GO TO 8401                                          04080000
      IF(NODE3.LE.0)GO TO 8401                                          04090000
      IF(TS.LT.0.)GO TO 8438                                            04100000
      IF(TX.LT.0.)GO TO 8438                                            04110000
      IF(TY.LT.0.)GO TO 8438                                            04120000
      IF(E.LE.0.)GO TO 8416                                             04130000
      IF(G.LE.0.)GO TO 8417                                             04140000
      IF(X(NODE1).EQ.0.)GO TO 8419                                      04150000
      IF(X(NODE2).EQ.0.)GO TO 8419                                      04160000
      IF(X(NODE3).EQ.0.)GO TO 8419                                      04170000
      NSMALL = MIN0(NODE1,NODE2,NODE3)                                  04180000
      NLARGE = MAX0(NODE1,NODE2,NODE3)                                  04190000
      IF(IDNO.EQ.4)GO TO 210                                            04200000
      NODE4=NODE4-NBASE+1                                               04210000
      IF(NODE4.EQ.0)GO TO 8446                                          04220000
      IF(X(NODE4).EQ.0.)GO TO 8419                                      04230000
      IF(NODE4.LT.NSMALL) NSMALL = NODE4                                04240000
      IF(NODE4.GT.NLARGE) NLARGE = NODE4                                04250000
  210 CONTINUE                                                          04260000
      NB1 = INDEX(NLARGE+1) - INDEX(NSMALL)                             04270000
      IF(NB1.LE.NBAND)GO TO 213                                         04280000
      IF(NB1.LE.MXBAND) GO TO 211                                       04290000
      NOGO = 1                                                          04300000
      WRITE(6,115) NCARD                                                04310000
      GO TO 213                                                         04320000
  211 NBAND = NB1                                                       04330000
  213 IF(IDNO.EQ.4)GO TO 216                                            04340000
      IF(NOGO-1)215,12,215                                              04590000
```

```
MEMBER NAME  INPUT681
  215 IF(NOGO.EQ.1)GO TO 12                                                  04610000
      WRITE( 4)IDNO,NODE1,NODE2,NODE3,NODE4,IS,IX,IY,E,G,X(NODE1),           04620000
     1Y(NODE1),Z(NODE1),X(NODE2),Y(NODE2),Z(NODE2),X(NODE3),Y(NODE3),        04630000
     2Z(NODE3),X(NODE4),Y(NODE4),Z(NODE4),                                   04640000
     7KONEQ(NODE1),KONEQ(NODE2),KONEQ(NODE3),KONEQ(NODE4),INDEX(NODE1),      04650000
     9 INDEX(NODE2),INDEX(NODE3),INDEX(NODE4),(ITEMP(I),I=31,68)            04660000
      GO TO 12                                                               04670000
  216 IF(NOGO.EQ.1)GO TO 12                                                  04690000
  220 WRITE( 4)IDNO,NODE1,NODE2,NODE3,IS,IX,IY,E,G,X(NODE1),Y(NODE1),        04700000
     1Z(NODE1),X(NODE2),Y(NODE2),Z(NODE2),X(NODE3),Y(NODE3),Z(NODE3),        04710000
     2KONEQ(NODE1),KONEQ(NODE2),KONEQ(NODE3),INDEX(NODE1),INDEX(NODE2),      04720000
     3 INDEX(NODE3),(ITEMP(I),I=25,68)                                       04730000
      GO TO 12                                                               04740000
  217 IF(NOGO.EQ.1) GO TO 12                                                 04760000
      WRITE( 4)IDNO,NODE1,NODE2,AXAREA,BIYY,AYY,BIZZ,AZZ,BJ,NFIX1,NFIX2,04770000
     1NFIX3,NFIX4,NFIX5,NFIX6,E,G,X(NODE1),Y(NODE1),Z(NODE1),XA(NODE1),      04780000
     2YA(NODE1),ZA(NODE1),X(NODE2),Y(NODE2),Z(NODE2),XA(NODE2),YA(NODE2)04790000
     3,ZA(NODE2),KONEQ(NODE1),KONEQ(NODE2),INDEX(NODE1),INDEX(NODE2),        04800000
     4          (ITEMP(I),I=34,68)                                           04810000
      GO TO 12                                                               04820000
  218 IF(NOGO.EQ.1) GO TO 12                                                 04840000
      IF(NODE3.LT.0)GO TO 8401                                               04850000
      WRITE( 4)IDNO,NODE1,NODE2,NODE3,MCODE,AXAREA,BIYY,AYY,BIZZ,AZZ,BJ,04860000
     1NFIX1,NFIX2,NFIX3,NFIX4,NFIX5,NFIX6,E,G,X(NODE1),Y(NODE1),Z(NODE1)04870000
     2,X(NODE2),Y(NODE2),Z(NODE2),X(NODE3),Y(NODE3),Z(NODE3),XA(NODE3),      04880000
     3YA(NODE3),ZA(NODE3),                                                   04890000
     4 KONEQ(NODE1),KONEQ(NODE2),INDEX(NODE1),INDEX(NODE2),                  04900000
     5          (ITEMP(I),I=36,68)                                           04910
      GO TO 12                                                               04920000
  219 IF(NOGO.EQ.1) GO TO 12                                                 04940000
      WRITE( 4)IDNO,NODE1,NODE2,AXAREA,E,X(NODE1),Y(NODE1),Z(NODE1),         04950000
     1X(NODE2),Y(NODE2),Z(NODE2),                                            04960000
     2KONEQ(NODE1),KONEQ(NODE2),INDEX(NODE1),INDEX(NODE2),(ITEMP(I),I=       04970000
     3   16,68)                                                              04980000
      GO TO 12                                                               04990000
  130 FORMAT(2X,19A4,A2)                                                     05000000
  132 FORMAT(2X,19A4,A2,/////,25X,'***   I N P U T   L I S T I N G   ***',05000100
     1///////////)                                                           05001000
  129 FORMAT(1H1)                                                            05010000
  100 FORMAT(6I4)                                                            05020000
  500 FORMAT(//25X,'***   C O N T R O L   I N P U T   ***',//)               05030000
  501 FORMAT(20X,30HLOWEST NODE NUMBER DESIGNATION, 7X,I3//                  05040000
     120X,26HNUMBER OF STRUCTURAL NODES,10X,I4//                             05050000
     220X,17HNUMBER OF A-NODES,19X,I4//                                      05060000
     320X,21HNUMBER OF LOAD CONDS,,15X,I4)                                   05070000
  505 FORMAT(1H1,/45X,'*** C O O R D I N A T E   I N P U T ***',//)          05080000
  507 FORMAT(11X,8HNODE NO.,19X,1HX,15X,1HY,15X,1HZ,12X,'EQUATION CONTRO05090000
     1L',10X,5HINDEX,/)                                                      05095000
  101 FORMAT(1H1//10X,45HE R R O R  -  THE NUMBER OF A-NODE CARDS READ, 05100000
     13H IS,I4)                                                              05110000
  102 FORMAT(1H1//10X,44HE R R O R  -  THE NUMBER OF STRUCTURAL NODE ,       05120000
     113HCARDS READ IS,I4)                                                   05130000
  107 FORMAT(3F11.6)                                                         05140000
  506 FORMAT( I15,1X,A1,12X,3E16.5,/)                                        05150000
  508 FORMAT(1H1,45X,'***  E L E M E N T   I N P U T  ***'//)                05152000
  509 FORMAT(1H1,45X,'***  L O A D S   I N P U T  ***'//)                    05154000
  510 FORMAT(I15,14X,3E16.5,4X,6(2X,I1),10X,I5,/)                            05160000
  111 FORMAT(I2,19A4,A2)                                                     05170000
  112 FORMAT(3I4,A1,6I8,3X,6I1,2E5.1)                                        05180000
  120 FORMAT(/2I7,1X,3I5,2X,A1,2X,6E10.3,2X,6I1,2X,2E12.3)                   05190000
  127 FORMAT (//,3X,'ELEM NO',2X,'ID',3X,'N1',3X,'N2',3X,'N3',9X,'AREA',05192000
     17X,'IY',8X,'AY',8X,'IZ',8X,'AZ',8X,'J',8X,'PINS' ,10X,'E',11X,'G')05193000
  123 FORMAT (//,3X,'ELEM NO',2X,'ID',3X,'N1',3X,'N2',14X,'AREA',18X,        05194000
     1'E')                                                                   05195000
  125 FORMAT(/2I7,1X,2I5,11X,F10.3,11X,F12.3)                                05196000
  121 FORMAT(/20X,5HDIST=,E14.7)                                             05200000
 9401 FORMAT(/10X,35HONE OF THE NODE NUMBERS IS IN ERROR/)                   05210000
 9404 FORMAT(/10X,26HTHE AXIAL AREA IS IN ERROR/)                            05220000
 9405 FORMAT(/10X,15HIYY IS NEGATIVE/)                                       05230000
 9406 FORMAT(/10X,15HAYY IS NEGATIVE/)                                       05240000
```

```
 9407 FORMAT(/10X,15HIZZ IS NEGATIVE/)                                  05250000
 9408 FORMAT(/10X,15HAZZ IS NEGATIVE/)                                  05260000
 9409 FORMAT(/10X,14HJ IS NEGATIVE/)                                    05270000
 9416 FORMAT(/10X,13HE IS NEGATIVE/)                                    05280000
 9417 FORMAT(/10X,13HG IS NEGATIVE/)                                    05290000
 9419 FORMAT(/10X,41HONE OF THE X COORDINATES IS EQUAL TO ZERO/)        05310000
 9421 FORMAT(/10X,40HTHERE IS AN ILLEGAL CHARACTER IN COL. 15/)         05320000
 9422 FORMAT(/10X,37HTHE FIXITY CONTROLS FOR X ARE UNEQUAL/)            05330000
 9423 FORMAT(/10X,43HIYY AND AYY MUST BE POSITIVE OR THE FIXITY         05340000
     124HCONTROLS FOR Y MUST BE 1/)                                     05350000
 9424 FORMAT(/10X,43HIZZ AND AZZ MUST BE POSITIVE OR THE FIXITY         05360000
     124HCONTROLS FOR Z MUST BE 1/)                                     05370000
 9427 FORMAT(/10X,45HIF J=0 THE FIXITY CONTROLS FOR X MUST EQUAL 1)     05380000
 9430 FORMAT(/10X,31HTS, TX, AND TY MUST BE POSITIVE/)                  05410000
 9446 FORMAT(/10X,23HNODE 4 IS EQUAL TO ZERO/)                          05420000
  114 FORMAT(A14,3F7.3,2E5.1)                                           05430000
  122 FORMAT(/2I7,1X,4I5,5X,E10.3,10X,E10.3,10X,F10.3,5X,2E10.3)        05440000
 7000 FORMAT(/2I7,1X,3I5,10X,E10.3,10X,E10.3,10X,F10.3,5X,2E10.3)       05441000
 7002 FORMAT (//,3X,'ELEM NO',2X,'ID',3X,'N1',3X,'N2',3X,'N3',15X,'TS', 05442000
     118X,'TX',18X,'TY',15X,'E',9X,'G')                                 05443000
 7004 FORMAT (//,3X,'ELEM NO',2X,'ID',3X,'N1',3X,'N2',3X,'N3',3X,'N4',  05444000
     110X,'TS',18X,'TX',18X,'TY',15X,'E',9X,'G')                        05445000
  115 FORMAT(/5X,10H**********,3X,27HTHE SEPARATION ON CARD NO. ,I5,    05450000
     113H IS TOO LARGE)                                                 05460000
      DIMENSION ARLDS(6)                                                05500000
 2000 WRITE(6,509)                                                      05510000
 2266 FORMAT(13X,   8H    NODE,5X,4HCOND,91X,8HCARD NO./)               05530000
      WRITE(6,2266)                                                     05540000
 2001 READ(IN,1103) NODEE,(DUMMY(I),I=1,19)                             05550000
      IF(NODEE.NE.0)   GO TO 3300                                       05560000
      WRITE(6,1213) (DUMMY(I),I=1,19)                                   05570000
      GO TO 2200                                                        05570100
 3300 WRITE(0,91476)(DUMMY(I),I=1,19)                                   05570300
      READ(0,2002) KONDNO,ARLDS                                         05570310
 2002 FORMAT(    I4,6F12.0)                                             05570400
      NREC11 = NREC11 + 1                                               05580000
      IF( (NODEE.GE.NBASE).AND.(NODEE.LE.(NSTNDS+NBASE)).AND.           05590000
     1(KONDNO.GE.1).AND.(KONDNO.LE.LSTCON) ) GO TO 2100                 05600000
      NOGO = 1                                                          05610000
 6602 FORMAT(1X 11H** ERROR **,2I9,6E15.6,I9)                           05620000
      WRITE(6,6602) NODEE, KONDNO, ARLDS, NREC11                        05630000
      GO TO 2001                                                        05640000
 2100 WRITE(6,6601) NODEE, KONDNO, ARLDS, NREC11                        05650000
      NODEE = NODEE -NBASE + 1                                          05660000
      IF(NOGO.EQ.1) GO TO 2001                                          05670000
      WRITE(11) NODEE, KONDNO, ARLDS                                    05680000
 6601 FORMAT(12X2I9,6E15.6,I9)                                          05690000
      GO TO 2001                                                        05700000
 2200 IF(NOGO.NE.1) WRITE(6,2202)                                       05710000
 2202 FORMAT( //5X26HNO ERRORS IN LOADS CARDS..//)                      05720000
      RETURN                                                            05730000
      END                                                               05740000
```

```
MEMBER NAME  LOADG81
      SUBROUTINE LOADER(ARRAY,MODE)                                     00010000
      DIMENSION CARD(5),ARRAY(1),AREA(20)                               00050000
      COMMON / PART / TITLE(20)                                         00060000
      GO TO (100,200),MODE                                              00080000
  100 READ(15,401) AREA                                                 00090000
  401 FORMAT(20A4)                                                      00090100
      WRITE(2,401) AREA                                                 00090200
      READ(5,101) NSUB,NOPC,CARD                                        00090300
  101 FORMAT(I4,I1,5E14.7)                                              00100000
      IF(NSUB.LE.0) GO TO 303                                           00110000
      IF(NOPC.LE.0) GO TO 303                                           00120000
      IF(NOPC-6)109,303,108                                             00130000
  108 IF(NOPC-8)102,111,112                                             00140000
  109 DO 110 JCARD=1,NOPC                                               00150000
      JARRAY=NSUB + JCARD - 1                                           00160000
  110 ARRAY(JARRAY) = CARD(JCARD)                                       00170000
      GO TO 100                                                         00180000
  102 READ(15,401) AREA                                                 00190000
      WRITE(2,401) AREA                                                 00190100
      READ(5,103) TITLE                                                 00190200
  103 FORMAT(20A4)                                                      00200000
  104 NPAGE = 0                                                         00230000
      GO TO 100                                                         00240000
  111 RETURN                                                            00250000
  112 STOP                                                              00260000
  200 NPAGE = NPAGE + 1                                                 00270000
      WRITE(6,201) (TITLE(N),N=1,20),NPAGE                              00280000
  201 FORMAT(1H1,20A4,10X,6H PAGE ,I3,//)                               00290000
      RETURN                                                            00300000
  303 WRITE(6,304)NSUB,NOPC,ARRAY                                       00310000
  304 FORMAT()9H INVALID DATA CARD ,I4,I1,5E14.7)                       00320000
      STOP                                                              00330000
      END                                                               00340000

MEMBER NAME  MAING81
      COMMON NELREC, NREC5, NREC11, LSTCON, NBAND, MXBAND,              00040000
     1        MXLCND, MXNODS, NBASE, MSTIFF, MSTRES, LIN,               00050000
     2        NDFREC,NOCO,NSTRES,KNILOS,MUDDY1,MUDDY2                   00060000
      COMMON/RCNT/JRECS                                                 00070000
      COMMON/JULY/IONE                                                  00090000
      COMMON/BLOCK5/ ITEMP(68)                                          00100000
      CALL FRRSET(209,300,-1)                                           00120000
    2 REWIND 4                                                          00130000
      REWIND 8                                                          00140000
      REWIND 9                                                          00150000
      REWIND 11                                                         00160000
      CALL DATA                                                         00170000
      CALL INPUT                                                        00180000
       REWIND 11                                                        00190000
      CALL GENRTN                                                       00271000
      CALL EXIT                                                         00272000
      END                                                               00280000

MEMBER NAME  DIMU2G81
      SUBROUTINE MATMU2(A,B,C,NRA,NCA,NCB)                              00010000
      DIMENSION A(24,24),B(24,24),C(24,24),TEMP(24)                     00020000
      DOUBLE PRECISION D,E,F                                            00030000
      DO 3 I=1,NRA                                                      00040000
      DO 2 J=1,NCB                                                      00050000
      D=0.                                                              00060000
      DO 1 K=1,NCA                                                      00070000
      E=A(I,K)                                                          00080000
      F=B(K,J)                                                          00090000
    1 D=D+E*F                                                           00100000
    2 TEMP(J)=D                                                         00110000
      DO 3 KK=1,NCB                                                     00120000
    3 C(I,KK)=TEMP(KK)                                                  00130000
      RETURN                                                            00140000
      END                                                               00150000
```

MEMBER NAME TRANGGP1

```
      SUBROUTINE TRIANG
      COMMON /BLOCK5/  ITEMP(68)
      COMMON/BLOCK2/          NX(24),S1(24,24),S2(24,24),TRANS2(24,24),
     1TRANS3(24,24),U1(24,24),U2(24,24),U3(24,24),U4(24,24),U5(24,24)
     2,NODE(4)
      DIMENSION TEMP(68),TRANS(3,3)
      DIMENSION INDEX(4)
      EQUIVALENCE (TEMP(1),ITEMP(1))
      NODE(1)=ITEMP(2)
      NODE(2)=ITEMP(3)
      NODE(3)=ITEMP(4)
      TS=TEMP(5)
      TX=TEMP(6)
      TY=TEMP(7)
      E=TEMP(8)
      G=TEMP(9)
      X1=TEMP(10)
      Y1=TEMP(11)
      Z1=TEMP(12)
      X2=TEMP(13)
      Y2=TEMP(14)
      Z2=TEMP(15)
      X3=TEMP(16)
      Y3=TEMP(17)
      Z3=TEMP(18)
      CALL UNPAK2(ITEMP(19), NX(1))
      CALL UNPAK2(ITEMP(20), NX(7))
      CALL UNPAK2(ITEMP(21), NX(13))
      INDEX(1)=ITEMP(22)
      INDEX(2)=ITEMP(23)
      INDEX(3)=ITEMP(24)
      DO 20 J=1,24
      DO 20 I=1,24
      S1(I,J)=0.
      S2(I,J)=0.
      TRANS2(I,J)=0.
   20 TRANS3(I,J)=0.
      DO 24 J=1,3
      DO 24 I=1,3
   24 TRANS(I,J)=0.
      DETA=Y2*Z3+Y1*Z2+Z1*Y3-(Y2*Z1+Y3*Z2+Y1*Z3)
      DETB=X1*Z3+Z2*X3+Z1*X2-(X3*Z1+X1*Z2*X2*Z3)
      DETC=X1*Y2+Y1*X3+X2*Y3-(X3*Y2+Y3*X1+X2*Y1)
      GG=SQRT(DETA**2+DETB**2+DETC**2)
      D=SQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)
      TRANS(1,3)=DETA/GG
      TRANS(2,3)=DETB/GG
      TRANS(3,3)=DETC/GG
      TRANS(1,1)=(X2-X1)/D
      TRANS(2,1)=(Y2-Y1)/D
      TRANS(3,1)=(Z2-Z1)/D
      TRANS(1,2)=TRANS(2,3)*TRANS(3,1)-TRANS(2,1)*TRANS(3,3)
      TRANS(2,2)=TRANS(3,3)*TRANS(1,1)-TRANS(3,1)*TRANS(1,3)
      TRANS(3,2)=TRANS(1,3)*TRANS(2,1)-TRANS(1,1)*TRANS(2,3)
      BX1=X1*TRANS(1,1)+Y1*TRANS(2,1)+Z1*TRANS(3,1)
      BY1=X1*TRANS(1,2)+Y1*TRANS(2,2)+Z1*TRANS(3,2)
      BZ1=X1*TRANS(1,3)+Y1*TRANS(2,3)+Z1*TRANS(3,3)
      BX2=X2*TRANS(1,1)+Y2*TRANS(2,1)+Z2*TRANS(3,1)
      BY2=X2*TRANS(1,2)+Y2*TRANS(2,2)+Z2*TRANS(3,2)
      BZ2=X2*TRANS(1,3)+Y2*TRANS(2,3)+Z2*TRANS(3,3)
      BX3=X3*TRANS(1,1)+Y3*TRANS(2,1)+Z3*TRANS(3,1)
      BY3=X3*TRANS(1,2)+Y3*TRANS(2,2)+Z3*TRANS(3,2)
      BZ3=X3*TRANS(1,3)+Y3*TRANS(2,3)+Z3*TRANS(3,3)
      TSQ=TS*TS
      X32=BX3-BX2
      X13=BX1-BX3
      X21=BX2-BX1
      Y23=BY2-BY3
      Y31=BY3-BY1
      Y12=BY1-BY2
      ONED2A=1./(BY1*X32+BY2*X13+BY3*X21)
      AMU = (E/(2.*G)) - 1.
      GX = AMU
```

```
      GY = AMU * TX/TY                                                  00843000
      IF(TX.LT.TY) GO TO 30                                             00844000
      GX = AMU * TY/TX                                                  00845000
      GY = AMU                                                          00846000
   30 CONTINUE                                                          00847000
      E = E / (1.-GX*GY)                                                00848000
      AREA = (BY1 * X32 + BY2 * X13 + BY3 * X21)/2.
      VOLUME = TS * AREA
      CONST = SQRT((AREA*TX)/(2.*E))                                    00849E00
      S1(1,1)=E*Y23                                                     00850000
      S1(1,2)=E*GX*X32                                                  00860000
      S1(1,3)=E*Y31                                                     00870000
      S1(1,4)=E*GX*X13                                                  00880000
      S1(1,5)=E*Y12                                                     00890000
      S1(1,6)=E*GX*X21                                                  00900000
      DO 40 I=1,6                                                       00901000
   40 S1(1,I) = S1(1,I) * CONST                                         00902000
      CONST = SQRT((AREA*TY)/(2.*E))                                    00903000
      S1(2,1)=E*GY*Y23                                                  00910000
      S1(2,2)=E*X32                                                     00920000
      S1(2,3)=E*GY*Y31                                                  00930000
      S1(2,4)=E*X13                                                     00940000
      S1(2,5)=E*GY*Y12                                                  00950000
      S1(2,6)=E*X21                                                     00960000
      DO 50 I=1,6                                                       00961000
   50 S1(2,I) = S1(2,I) * CONST                                         00962000
      CONST=SQRT(AREA*TS)/(2.0*G)                                       00963000
      S1(3,1)=G*X32                                                     00970000
      S1(3,2)=G*Y23                                                     00980000
      S1(3,3)=G*X13                                                     00990000
      S1(3,4)=G*Y31                                                     01000000
      S1(3,5)=G*X21                                                     01010000
      S1(3,6)=G*Y12                                                     01020000
      DO 60 I=1,6                                                       01021000
   60 S1(3,I) = S1(3,I) * CONST                                         01022000
      S1(4,1)=TSG*X32                                                   01030000
      S1(4,2)=TSG*Y23                                                   01040000
      S1(4,3)=TSG*X13                                                   01050000
      S1(4,4)=TSG*Y31                                                   01060000
      S1(4,5)=TSG*X21                                                   01070000
      S1(4,6)=TSG*Y12                                                   01080000
      DO 1 J=1,6                                                        01100000
      DO 1 I=1,4                                                        01110000
    1 S1(I,J)=S1(I,J)*DNEO2A                                            01120000
      S2(1,1)=Y23**2*TX*E+X32**2*TSG                                    01170000
      S2(2,1)=X32*Y23*(GY*TY*E+TSG)                                     01180000
      S2(2,2)=X32**2*TY*E+Y23**2*TSG                                    01190000
      S2(3,1)=Y31*Y23*TX*E+X13*X32*TSG                                  01200000
      S2(3,2)=Y31*X32*GX*TX*E+X13*Y23*TSG                               01210000
      S2(3,3)=Y31**2*TX*E+X13**2*TSG                                    01220000
      S2(4,1)=X13*Y23*GY*TY*E+Y31*X32*TSG                               01230000
      S2(4,2)=X13*X32*TY*E+Y31*Y23*TSG                                  01240000
      S2(4,3)=Y31*X13*(GY*TY*E+TSG)                                     01250000
      S2(4,4)=X13**2*TY*E+Y31**2*TSG                                    01260000
      S2(5,1)=Y12*Y23*TX*E+X21*X32*TSG                                  01270000
      S2(5,2)=Y12*X32*GY*TY*E+X21*Y23*TSG                               01280000
      S2(5,3)=Y12*Y31*TX*E+X21*X13*TSG                                  01290000
      S2(5,4)=Y12*X13*GX*TX*E+X21*Y31*TSG                               01300000
      S2(5,5)=Y12**2*TX*E+X21**2*TSG                                    01310000
      S2(6,1)=X21*Y23*GY*TY*E+Y12*X32*TSG                               01320000
      S2(6,2)=X21*X32*TY*E+Y12*Y23*TSG                                  01330000
      S2(6,3)=X21*Y31*GY*TY*E+Y12*X13*TSG                               01340000
      S2(6,4)=X21*X13*TY*E+Y12*Y31*TSG                                  01350000
      S2(6,5)=X21*Y12*(GY*TY*E+TSG)                                     01360000
      S2(6,6)=X21**2*TY*E+Y12**2*TSG                                    01370000
      DO 2 J=2,6                                                        01410000
      K=J-1                                                             01420000
      DO 2 I=1,K                                                        01430000
    2 S2(I,J)=S2(J,I)                                                   01440000
      DNEO4A=.5*DNEO2A                                                  01480000
```

```
      DO 3 J=1,6                                                        01490000
      DO 3 I=1,6                                                        01500000
    3 S2(I,J)=S2(I,J)+ONLOAA                                            01510000
      TRANS2(1,4)=TRANS(1,1)                                            01530000
      TRANS2(1,5)=TRANS(2,1)                                            01540000
      TRANS2(1,6)=TRANS(3,1)                                            01550000
      TRANS2(2,4)=TRANS(1,2)                                            01560000
      TRANS2(2,5)=TRANS(2,2)                                            01570000
      TRANS2(2,6)=TRANS(3,2)                                            01580000
      TRANS2(3,10)=TRANS(1,1)                                           01590000
      TRANS2(3,11)=TRANS(2,1)                                           01600000
      TRANS2(3,12)=TRANS(3,1)                                           01610000
      TRANS2(4,10)=TRANS(1,2)                                           01620000
      TRANS2(4,11)=TRANS(2,2)                                           01630000
      TRANS2(4,12)=TRANS(3,2)                                           01640000
      TRANS2(5,16)=TRANS(1,1)                                           01650000
      TRANS2(5,17)=TRANS(2,1)                                           01660000
      TRANS2(5,18)=TRANS(3,1)                                           01670000
      TRANS2(6,16)=TRANS(1,2)                                           01680000
      TRANS2(6,17)=TRANS(2,2)                                           01690000
      TRANS2(6,18)=TRANS(3,2)                                           01700000
      DO 4 J=1,6                                                        01710000
      DO 4 I=1,18                                                       01720000
    4 TRANS3(I,J)=TRANS2(J,I)                                           01730000
      CALL MATMU2(S1,TRANS2,S1,4,6,18)                                  01760000
      CALL MATMU2(TRANS3,S2,TRANS3,18,6,6)                              01770000
      CALL MATMU2(TRANS3,TRANS2,TRANS3,18,6,18)                         01780000
      CALL BANDS(TRANS3,S1,18,4,NX,INDEX,NODE,VOLUME)                   01810000
      RETURN                                                            01820000
      END                                                               01830000
```

MEMBER NAME UNPAK681

```
      SUBROUTINE UNPAK(NUMPKD,IUNPKD,KSAVE)                             00010000
      DIMENSION IUNPKD(6)                                               00080000
      KSAVE = 0                                                         00090000
      K = NUMPKD                                                        00100000
      DO 15 L=1,6                                                       00110000
      I = 6 - L                                                         00120000
      NED = 10 ** I                                                     00130000
      IUNPKD(L) = K / NED                                               00140000
      IF( IUNPKD(L).EQ.0) KSAVE = KSAVE + 1                             00150000
      K = K - NED * IUNPKD(L)                                           00160000
   15 CONTINUE                                                          00170000
      RETURN                                                            00180000
      END                                                               00190000
```

MEMBER NAME UNPK2681

```
      SUBROUTINE UNPAK2(NUMPKD,IUNPKD)                                  00010000
      DIMENSION IUNPKD(6)                                               00060000
      K = NUMPKD                                                        00070000
      DO 15 L=1,6                                                       00080000
      I = 6 - L                                                         00090000
      NED = 10 ** I                                                     00100000
      IUNPKD(L) = K / NED                                               00110000
      K = K - NED * IUNPKD(L)                                           00120000
   15 CONTINUE                                                          00130000
      RETURN                                                            00140000
      END                                                               00150000
```

```
MEMBER NAME  ALD6B3
      SUBROUTINE ALD                                                    00010000
      DIMENSION KDUM(155)                                               00020000
      DIMENSION SCOL(300,6),AELMTX(6,6),KONEQ(1500),                    00030000
     1XYZ(100,6),XYZSUB(6), KONTRL(6)                                   00040000
      COMMON NELREC, MRECS, NRECII, LSTCON, NBAND, MXBAND,              00050000
     1       MXCLND, MXNODS, NBASE, MSTIFF, MSTRES, LIN,                00060000
     2       NSFREC, NOGO, NSTRPS, KNTLDS, MUDDY1, MUDDY2               00070000
      DIMENSION INFO(18)                                                00100000
      EQUIVALENCE (INFO(1),NELREC)                                      00110000
      READ ( 9) NZERO, NZERO, NZERO, LARR, NZERO, INFO                  00130000
    2 CONTINUE                                                          00180000
      READ( 9) NSINDS, I1, I2, LARR, J2, (KDUM(L),L=1,J2)               00190000
      LX = I2 + 1                                                       00200000
      DO 4 L=1,J2                                                       00210000
      LX=LX-1                                                           00220000
      KONEQ(LX) = KDUM(L)                                               00230000
    4 CONTINUE                                                          00240000
      IF(I1.NE.1)GO TO 2                                                00250000
      KOUNT = 100                                                       00270000
      MLINES = 54                                                       00280000
      KNTLDS = LIN - 1                                                  00290000
 6600 FORMAT(1H1,10X,'PROGRAM CONTROL INFORMATION'///5X,                00300000
     1'NO. OF STRUC. NODES = ',I5/5X,'LOWEST NODE NO. = ',I4/5X,        00310000
     2'NO. OF LOAD CONDITIONS = ',I4/5X,'BANDWIDTH = ',I4/5X,           00320000
     3'NO. OF STRUCTURAL ELEMENTS = ',I6/5X,                            00330000
     4'ORDER OF STIFFNESS MATRIX IS ',I6)                               00330100
      WRITE(6,6600) NSINDS, NBASE, LSTCON, NBAND, MRECS, KNTLDS         00340000
      IROW1 = 1                                                         00490000
      ITRI = 1                                                          00500000
      NSTRPS = 0                                                        00510000
      KTRECS = 0                                                        00520000
       NB10=NBAND+10                                                    00530000
       JJ=1                                                             00540000
       LOWEST=1                                                         00550000
    5 CONTINUE                                                          00560000
      DO 10 J=1,6                                                       00610000
       DO 10 I=1,NB10                                                   00620000
   10 SCOL(I,J) = 0.                                                    00630000
       JSAVE=JJ                                                         00640000
       IF(JSAVE.NE.1) GO TO 20                                          00650000
   15 IF(KTRECS.EQ.NSFREC)GO TO(80,90),ITRI                             00660000
      KTRECS = KTRECS + 1                                               00680000
      READ( 8) II, JJ, NROWS, NCOLS,((AELMTX(I1,J1),J1=1,NCOLS),        00700000
     1 I1=1,NROWS)                                                      00710000
   20  IF(JJ.NE.JSAVE) GO TO (80,90),ITRI                               00770000
       IF(II.GT.JSAVE.OR.II.LT.LOWEST) GO TO 15                         00810000
       I=II-LOWEST                                                      00860000
      DO 25 I1 = 1,NROWS                                                00870000
      I = I + 1                                                         00880000
      DO 25 J = 1,NCOLS                                                 00890000
   25 SCOL(I,J) = SCOL(I,J) + AELMTX(I1,J)                              00900000
      MCOL = NCOLS                                                      00910000
      GO TO 15                                                          00920000
   80 CONTINUE                                                          00960000
      DO 85 K =1,MCOL                                                   01000000
      NSTRPS = NSTRPS + 1                                               01010000
      IF(NSTRPS.LE.NBAND) ILOW = 1                                      01020000
       WRITE( 1)(SCOL(I,K),I=ILOW,NSTRPS)                               01030000
       ILOW=ILOW+1                                                      01050000
      IF(ILOW.GT.6) LOWEST = 2                                          01060000
   85 CONTINUE                                                          01070000
       IF(NSTRPS.GE.NBAND) ITRI=2                                       01110000
      IF( KTRECS.EQ.NSFREC )GO TO 150                                   01120000
      GO TO 5                                                           01130000
   90 CONTINUE                                                          01170000
       IROW1=ILOW-LOWEST                                                01200000
      DO 100 K = 1,MCOL                                                 01210000
       NSTRPS = NSTRPS +1                                               01220000
      IEND = IROW1 + NBAND                                              01230000
      IROW1 = IROW1 + 1                                                 01240000
```

```
      WRITE(1 )(SFOL(I,K),I=IROW1,IEND)                                  01250000
  100 CONTINUE                                                           01270000
      IF(NSFREC.EQ.KTRECS)GO TO 150                                      01280000
       ILOW=ILOW+MCOL                                                    01290000
       LOWEST=ILOW-5                                                     01300000
       IF(LOWEST.LT.1)LOWEST=1                                           01310000
       GO TO 5                                                           01320000
  150 CONTINUE                                                           01390000
      REWIND  8                                                          01400000
      REWIND 1                                                           01410000
      IND = 0                                                            02090000
      KNT = 0                                                            02100000
      NOD = 1                                                            02110000
  210 CONTINUE                                                           02180000
      DO 215 I1 =1,LSTCON                                                02190000
      DO 215 J1 =1,6                                                     02200000
  215 XYZ(I1,J1) = 0.                                                    02210000
      IF(KNT)220,220,225                                                 02220000
  220 READ(11) NSUB,KSUB,XYZSUB                                          02240000
      KNT = KNT + 1                                                      02250000
  225 CONTINUE                                                           02280000
      IF(NSUB.NE.NOD) GO TO 235                                          02290000
      DO 230 J1=1,6                                                      02300000
  230 XYZ(KSUB,J1) = XYZSUB(J1)                                          02310000
      IF(KNT.LT.NREC11)GO TO 220                                         02320000
  235 CONTINUE                                                           02330000
      CALL UNPAK2(KONEQ(NOD),KONTRL)                                     02340000
      DO 240 I1=1,6                                                      02370000
      IF(KONTRL(I1).NE.0)GOTO 240                                        02380000
      IND = IND + 1                                                      02390000
      WRITE( 3)IND,(XYZ(K,I1),K=1,LSTCON)                                02400000
  240 CONTINUE                                                           02480000
      NOD = NOD + 1                                                      02490000
      IF(KNT.LT.NREC11) GO TO 210                                        02500000
       IF(IND.LT.NSTRPS) GO TO 210                                       02510000
      REWIND 11                                                          02520000
      REWIND 3                                                           02530000
      RETURN                                                             02540000
      END                                                                02550000

MEMBER NAME  EXCH683
      SUBROUTINE EXCH(NBAND)                                             00010000
      COMMON/ATOS/S(42195),IROW, JY, JZ, KNT10                           00050000
      DO 110 J=2,NBAND                                                   00060000
      J2= J*+2                                                           00070000
      I1 = (J2 -3*J +2)/2 + 1                                            00080000
      I2 = (J2-J)/2                                                      00090000
      DO 110 I= I1, I2                                                   00100000
      IJ = I + J                                                         00110000
      S(I) = S(IJ)                                                       00120000
  110 CONTINUE                                                           00130000
      READ( 1) (S(I),I=JY,JZ)                                            00140000
      KNT10 = KNT10 + 1                                                  00150000
  900 RETURN                                                             00160000
      END                                                                00170000

MEMBER NAME  KAY683
      SUBROUTINE KAY                                                     00010000
      COMMON NFLREC, MRECS, NREC11, LSTCON, NHAND, MXBAND,               00020000
     1        MXCLND, MXNODS, NBASE, MSTIFF, MSTRES, LIN,                00030000
     2        NSFREC, NOGO, NSTRPS, KNTLDS, MUDDY1, MUDDY2               00040000
      COMMON/ATOS/GJUNK(42195),IROW,JY,JZ,KNT10
      DOUBLE PRECISION  SUM,TEMPA,TEMPB,TEMPC,TEMPD                      00105000
      DIMENSION  YAK(290,100), S(290), F(100)                            00110000
      EQUIVALENCE (GJUNK(1),YAK(1,1)) , (GJUNK(29001),S(1)) ,            00115000
     1 (GJUNK(29291),F(1))                                               00116000
      DO203I = 1,LSTCON                                                  00120000
      DO203J = 1,NBAND                                                   00130000
  203 YAK(J,I) = 0.                                                      00140000
```

```
      I = 0
      IBRNCH = 1
  205 I = I + 1
      IF(I.EQ.NBAND)IBRNCH = 2
  207 CONTINUE
      READ( 4)(S(I2),I2=1,I)
      READ( 3) MUD,(F(I3),I3=1,LSTCON)
      I1 = I - 1
      DO220 J = 1,LSTCON
      SUM = 0.
      IF(I1.LT.1)GO TO215
      DO210 L = 1,I1
      TEMPA = S(L)
      TEMPB = YAK(L,J)
  210 SUM = SUM + TEMPA * TEMPB
  215 TEMPC = F(J)
      TEMPD = S(I)
      YAK(I,J) = (TEMPC - SUM)/ TEMPD
  220 CONTINUE
      WRITE(10)MUD,(YAK(1,K),K=1,LSTCON)
  223 IF(MUD.EQ.KNTLDS) GO TO250
      GO TO (205,225),IBRNCH
  225 CONTINUE
      DO240 M1 =2,NBAND
      M2 = M1-1
      DO240 M3 =1,LSTCON
  240 YAK(M2,M3) = YAK(M1,M3)
      GO TO 207
  250 CONTINUE
      REWIND 4
      REWIND 3
      REWIND 10
  260 CONTINUE
      RETURN
      END
```

MEMBER NAME MAIN683

```
      COMMON NELREC, MRECS, NREC11, LSTCON, NBAND, MXBAND,
     1       MXCLND, MXNODS, NBASE, MSTIFF, MSTRES, LIN,
     2       NSFREC, NOGO, NSTRPS, KNTLDS, MUDDY1, MUDDY2
      REWIND 1
      REWIND 3
      REWIND  4
      REWIND 8
      REWIND 9
      REWIND  10
      REWIND 11
      REWIND  12
      CALL ALD
      CALL STFS
      CALL KAY
      CALL EXIT
      STOP 5
      END
```

MEMBER NAME STFS683

```
      SUBROUTINE STFS
      COMMON NELREC, MRECS, NREC11, LSTCON, NBAND, MXBAND,
     1       MXCLND, MXNODS, NBASE, MSTIFF, MSTRES, LIN,
     2       NSFREC, NOGO, NSTRPS, KNTLDS, MUDDY1, MUDDY2
      COMMON/ATOS/S(42195),IROW, JY, JZ, KNT10
      DOUBLE PRECISION  SUM,TEMPA,TEMPB,TEMPC,SCOMB
      DIMENSION  SDUM(290),TEMPN(3,3),KILT(290)
      NBAND1 = NBAND + 1
      NBM1 = NBAND - 1
      IROW = 1
      NGUY = 0
      KOUNTR = 50
```

```
      CALL ERRSET(251,300,-1)
      CALL ERRSET(261,300,-1)
      I1 = 1
      DO 20 N =1,NBAND
      I2 = I1 + N - 1
      READ(1 )(S(I),I =I1,I2)
      I1 = I1 + N
      KILT(N) = 0
   20 CONTINUE
      JY = I1 - NBAND
      JZ = I1 - 1
      KNT10 = NBAND
      NBAK = 1
      S(1) = SQRT(S(1))
      SDUM (1) = S(1)
      K = 2
      DO 25 I=2,NBAND
      S(K) = S(K)/S(1)
      SDUM(I) = S(K)
      IF(SDUM(I).NE.0.)  KILT(I)=1
      K = K + 1
   25 CONTINUE
      WRITE(12) NBAK, (SDUM(I1),I1=1,NBAND)
      NOGUY = (NBAND-1) / 3
      NOGUY = (NBAND-1) -(NOGUY*3)
      DO 40 L = 2, NBAND
      ISUBLL=((L-(IROW-1))**2-(L-(IROW-1)))/2+1+L-IROW
      L1 = L-1
      DO 40 M = L, NBAND
      ISUBLM=((M-(IROW-1))**2-(M-(IROW-1)))/2+1+L-IROW
      IF(KILT(M).EQ.1)  GO TO 29
      IF(S(ISUBLM)) 29,37,29
   29 SUM = 0.
      DO 30 K = 1,L1
      ISUBKL=((L-(IROW-1))**2-(L-(IROW-1)))/2+1+K-IROW
      TEMPA = S(ISUBKL)
      ISUBKM=((M-(IROW-1))**2-(M-(IROW-1)))/2+1+K-IROW
      TEMPB = S(ISUBKM)
      SUM = SUM + TEMPA * TEMPB
   30 CONTINUE
      TEMPC = S(ISUBLM)
      SCOMB = TEMPC - SUM
      IF(L.NE.M) GO TO 35
      NGUY = NGUY + 1
      TEMPN(NGUY,1) = SCOMB
      TEMPN(NGUY,2) = S(ISUBLM)
      TEMPN(NGUY,3) = M
      IF(NGUY.LT.3)  GO TO 6969
      IF(KOUNTR.LT.50)  GO TO 6972
      WRITE(6,6970)
      KOUNTR = 1
 6972 WRITE(6,6971)((TEMPN(NG,NU),NU=1,3),NG=1,2)
      WRITE(6,4200)(TEMPN(3,NU),NU=1,3)
      KOUNTR=KOUNTR+2
      NGUY = 0
 6969 IF((L.EQ.NBAND).AND.(NOGUY.GT.0)) WRITE(6,6971)((TEMPN(NG,NU),NU=1
     1,3),NG=1,NOGUY)
      S(ISUBLM) = DSQRT(SCOMB)
      GO TO 38
   35 TEMPA = S(ISUBLL)
      S(ISUBLM) = SCOMB/TEMPA
   38 KILT(M) = 1
      GO TO 40
   37 S(ISUBLM) = 0.
   40 CONTINUE
      J1 = 1
      DO 60 I=1,NBAND
      J2 = J1 + I - 1
      WRITE( 4) (S(J),J=J1,J2)
      J1 = J1 + I
   60 CONTINUE
      IF(KNT10.EQ.NSTRPS) GO TO 150
      CALL EXCH(NBAND)
```

```
      M = NBAND
      NGUY = 0
      KOUNTR = 50
  100 CONTINUE
      IROW = IROW + 1
      M = M + 1
      KILL = 0
      IF(KNT10.LT.NSTRPS) GO TO 151
      NOGUY = (M-NBAND) / 3
      NOGUY = (M-NBAND) - (NOGUY*3)
  151 CONTINUE
      DO 115  L=IROW,M
      ISUBLM=((M-(IROW-1))**2-(M-(IROW-1)))/2+1+L-IROW
      IF(KILL.EQ.1) GO TO 103
      IF(S(ISUBLM)) 103,112,103
  103 SUM = 0.
      L1 = L-1
      IF(L1.LT.IROW) GO TO 108
      DO 105 K = IROW,L1
      ISUBKL=((L-(IROW-1))**2-(L-(IROW-1)))/2+1+K-IROW
      TEMPA = S(ISUBKL)
      ISUBKM=((M-(IROW-1))**2-(M-(IROW-1)))/2+1+K-IROW
      TEMPB = S(ISUBKM)
      SUM = SUM + TEMPA * TEMPB
  105 CONTINUE
  108 CONTINUE
      TEMPC = S(ISUBLM)
      SCOMB = TEMPC - SUM
      IF(L.NE.M)GO TO 110
      NGUY = NGUY + 1
      TEMPN(NGUY,1) = SCOMB
      TEMPN(NGUY,2) = S(ISUBLM)
      TEMPN(NGUY,3) = M
      IF(NGUY.LT.3)  GO TO 7069
      IF(KOUNTR.LT.50)  GO TO 7072
      WRITE(6,6970)
      KOUNTR = 1
 7072 WRITE(6,6971)((TEMPN(NG,NU),NU=1,3),NG=1,2)
      WRITE(6,4200)(TEMPN(3,NU),NU=1,3)
      KOUNTR=KOUNTR+2
      NGUY = 0
 7069 IF((L.EQ.M).AND.(NOGUY.GT.0).AND.(KNT10.EQ.NSTRPS))WRITE(6,6971)
     1 ((TEMPN(NG,NU),NU=1,3),NG=1,NOGUY)
      S(ISUBLM) = DSQRT(SCOMB)
      GO TO 111
  110 CONTINUE
      ISUBLL=((L-(IROW-1))**2-(L-(IROW-1)))/2+1+L-IROW
      TEMPA = S(ISUBLL)
      S(ISUBLM) = SCOMB/TEMPA
  111 KILL = 1
      GO TO 115
  112 S(ISUBLM) = 0.
  115 CONTINUE
      NBAK = NBAK + 1
      K=1
      DO 120 I=1,NBAND
      SDUM(I) = S(K)
      K=K+I
  120 CONTINUE
      WRITE(12) NBAK, (SDUM(I1),I1=1,NBAND)
      WRITE( 4) (S(K),K=JY,JZ)
      IF(KNT10.EQ.NSTRPS) GO TO 150
      CALL EXCH(NBAND)
      GO TO 100
  150 CONTINUE
      DO 180 I=2,NBAND
      DO 175 J=1,NBAND
      I1 = (J**2-J)/2 + 1
  175 SDUM(J) = S(I1)
      NBAK = NBAK + 1
```

```
      WRITE(12)NBAK,(SDUM(J1),J1=1,NBAND)
  180 CONTINUE
 6970 FORMAT('1',8X,'SII2',13X,'AII',8X,'ROW',7X,'SII2',13X,'AII',
     18X,'ROW')
 6971 FORMAT(1X,2(2E17.8,F5.0))
 4206 FORMAT(1X,2(2E17.8,F5.0))
      REWIND  4
      REWIND 12
      REWIND  1
      RETURN
      END
```

```
MEMBER NAME  UNPK2683
      SUBROUTINE UNPAK2(NUMPKD,IUNPKD)
      DIMENSION IUNPKD(6)
      K = NUMPKD
      DO 15 L=1,6
      I = 6 - L
      NED = 10 ** I
      IUNPKD(L) = K / NED
      K = K - NED * IUNPKD(L)
   15 CONTINUE
      RETURN
      END
```

```
MEMBER NAME  DEFL685
      SUBROUTINE DEFL
      REAL*8 XSV
      DIMENSION INFO(18)
      DIMENSION X(290,100),SC(290),YK2(100)
      DIMENSION XSV(6,100),INEQCN(6),KDUM(155)
      DOUBLE PRECISION SUM,TEMPA,TEMPB,TEMPC,TEMPD
      COMMON NELREC, MRECS, NREC11, LSTCON, NBAND, MXBAND,
     1        MXCLND, MXNODS, NBASE, MSTIFF, MSTRES, LIN,
     2        NSFREC, NOGO, NSTRPS, KNTLDS, MUDDY1, MUDDY2
      EQUIVALENCE (INFO(1),NELREC)
      READ( 9) NZERO, NZERO, NZERO, LARR, NZERO, INFO
      NSTNDS = NZERO
       KNTLDS= LIN-1
       NSTRPS = KNTLDS
 6600 FORMAT(1H1,10X,'PROGRAM CONTROL INFORMATION'///5X,
     1'NO. OF STRUC. NODES = ',I5/5X,'LOWEST NODE NO. = ',I4/5X,
     2'NO. OF LOAD CONDITIONS = ',I4/5X,'BANDWIDTH = ',I4/5X,
     3'NO. OF STRESS PACKAGES = ',I6/5X,
     4'NO. OF STIFF ROWS (COLS) AND NO. OF LOADS ROWS = ',I6)
      WRITE(6,6600) NSTNDS, NBASE, LSTCON, NBAND, MRECS, KNTLDS
      KOUNT=700
      MLINES=600
      DO 3 I = 1,LSTCON
      DO 3 J = 1,NBAND
    3 X(J,I) = 0.
      NUMDUM = LIN
      IBRNCH = 1
      I = 0
      NODE = NSTNDS + NBASE
      IREAD = 1
      GO TO 450
    5 I = I + 1
      IF(I.EQ.NBAND)IBRNCH = 2
```

```
    7 CONTINUE                                                          00430000
      READ(12) MUD, (SC(I2),I2=1,I)                                     00470000
      READ(10) MDU, (YK2(I3),I3=1,LSTCON)                               00510000
      DO 20 J = 1,LSTCON                                                00530000
      SUM = 0.                                                          00540000
      IF(I.EQ.1) GO TO 15                                               00550000
      DO 10 L = 2,I                                                     00560000
      L1 = I - L + 1                                                    00570000
      TEMPA = SC(L)                                                     00572000
      TEMPB = X(L1,J)                                                   00574000
   10 SUM = SUM + TEMPA * TEMPB                                         00580000
   15 TEMPC = YK2(J)                                                    00600000
      TEMPD = SC(1)                                                     00602000
      X(I,J) =  (TEMPC - SUM) / TEMPD                                   00604000
   20 CONTINUE                                                          00620000
      NUMDUM = NUMDUM - 1                                               00630000
      WRITE( 4) NUMDUM, (X(I,J1),J1=1,LSTCON)                           00670000
      NLINES = NLINES + 1                                               00680000
      IF(NLINES.NE.KEEPIT) GO TO 54                                     00690000
      DO 25 JK=1,6                                                      00700000
      DO 25 II=1,LSTCON                                                 00710000
   25 XSV(JK,II) = 0.                                                   00720000
       NSUB= I - KEEPIT                                                 00730000
      DO 32 KK=1,6                                                      00740000
      KKK = 7-KK                                                        00750000
      IF(INEGCN(KKK).GT.0)GO TO 32                                      00760000
       NSUB = NSUB + 1                                                  00770000
      DO 30 JJ = 1,LSTCON                                               00780000
   30 XSV(KKK,JJ) = X(NSUB,JJ)                                          00790000
   32 CONTINUE                                                          00800000
  613 FORMAT(1H1,37X,'J. SCIARRA'/,                                     00850000
     1              28X,'FINITE ELEMENT METHODOLOGY',/,                 00851000
     2              32X,'S-74 FINAL RESULTS',//,                        00852000
     3              24X,'ROTATIONS(RADIANS) AND DEFLECTIONS')           00853000
  611 FORMAT(//,2X,'NODE',I5,/,2X,'COND',4X,'ROTATION X',10X,           00870000
     1'ROTATION Y',10X,'ROTATION Z')                                    00880000
  614 FORMAT(8X,'DEFLECTION X',8X,'DEFLECTION Y',8X,'DEFLECTION Z')     00880100
  612 FORMAT(1X,I4,1PE17.7,1P2E20.7)                                    00890000
61212 FORMAT(1X,I4,1PE17.7,1P2E20.7)
      KIN = 1                                                           00900000
   38 KEND = KIN + 40                                                   00910000
      IF(KEND.GT.LSTCON) KEND = LSTCON                                  00920000
   39 KOUNT = KOUNT + KEND - KIN + 6                                    00930000
      IF(KOUNT.LE.MLINES) GO TO 401                                     00940000
      KOUNT = 0                                                         00950000
      WRITE(6,613)                                                      00960000
      GO TO 39                                                          00970000
  401 CONTINUE                                                          00980000
      WRITE(6,611) NODE                                                 00990000
      DO 40 KK = KIN,KEND                                               01000000
      WRITE(6,612)KK,(XSV(K1,KK),K1=1,3)                                01010000
      WRITE(11)  NODE, KK,(XSV(K1,KK),K1 =1,6)                          01011000
   40 CONTINUE                                                          01020000
      WRITE(6,614)                                                      01020100
      DO 41 KK=KIN,KEND                                                 01020200
      WRITE(6,61212)KK,(XSV(K1,KK),K1=4,6)
   41 CONTINUE                                                          01020400
      KIN = KEND + 1                                                    01030000
      IF(KIN.LE.LSTCON)GO TO 38                                         01040000
      IF(NUMDUM.EQ.1)GO TO 80                                           01050000
  450 CONTINUE                                                          01060000
      GO TO(455,460),IREAD                                              01100000
  455 CONTINUE                                                          01110000
      IREAD = 2                                                         01120000
```

```
      READ( 9) NSTNDS, I1, I2, LARR, JAKE, (KDUM(JK),JK=1,JAKE)
      LUG = 0
  460 CONTINUE
      LUG = LUG + 1
      IF(LUG.GT.JAKE) GO TO 455
      NODE = NODE - 1
      CALL UNPAK(KDUM(LUG),INEQCN,KEEPIT)
       IF(KEEPIT.LE.0) GO TO 460
      NLINES=0
   54 CONTINUE
      GO TO( 5, 55),IBRNCH
   55 CONTINUE
      DO 70 M1 = 2,NBAND
      M2 = M1-1
      DO 70 M3 = 1,LSTCON
   70 X(M2,M3) = X(M1,M3)
      GO TO 7
   80 CONTINUE
      REWIND 4
      IF(I1.NE.1) READ(9) NSTNDS,I1,I2,LARR,JAKE,(KDUM(JK),JK=1,JAKE)
      REWIND 10
      REWIND 12
      RETURN
      END
```

MEMBER NAME FSTR685

```
      SUBROUTINE FSTR
      DIMENSION FSS(100,28),STRESS(28,48),X(290,100)
     1, OPMAX(28), OPMIN(28), KONMAX(28), KONMIN(28)
     2,NODEAR(8), KLSBLK(8), INDNOS(8)
      COMMON NELREC, MRECS, NREC11, LSTCON, NBAND, MXBAND,
     1       MXCLND, MXNODS, NBASE, MSTIFF, MSTRES, LIN,
     2       NSFREC, NOGO, NSTRPS, KNTLDS, MUDDY1, MUDDY2
      DATA KMAX,KMIN /2HMX,2HMN /
      NSRECS = 0
      MLINES = 54
      MOST = MXBAND
      LIN1 = LIN-1
      IF(MXBAND.GT.LIN1) MOST= LIN1
      DO 4 I = 1,MOST
    4 READ(4) MUD,(X(I,J), J=1,LSTCON)
       INDLOW = LIN - MOST
    5 CONTINUE
      READ( 9) NUMREC, NBLKS, NRISM, (INDNOS(I1),I1=1,NBLKS), (NODEAR
     1(I1),I1=1,NBLKS), (KLSBLK(I1),I1=1,NBLKS),KOLTTL,
     2((STRESS(I1,J1),J1=1,KOLTTL),I1=1,NRISM),KIND,VOLUME
      NSRECS = NSRECS + 1
      IF(INDNOS(2).GE.INDLOW) GO TO 16
      IDIF = INDLOW - INDNOS(2)
      NEND = MOST - IDIF
      DO 10 I = 1,MOST
      IF(I.GT.NEND)GO TO 7
      I1 = I + IDIF
      DO 6 J = 1,LSTCON
    6 X(I,J) = X(I1,J)
      GO TO 10
    7 CONTINUE
      READ( 4) MUD, (X(I,J1),J1=1,LSTCON)
   10 CONTINUE
      INDLOW = INDLOW - IDIF
   16 CONTINUE
      DO 18 I = 1,NRISM
      OPMIN(I) = 1E10
   18 OPMAX(I) = -1.E10
      DO 25 J=1,NBLKS
```

```
      NODEAR(J) = NODEAR(J) + NBASE - 1                                 00850000
   25 CONTINUE                                                          00860000
      LLIN = 1                                                          00900000
      LLFIN = MLINES-14                                                 00910000
      IF( LSTCON .LT. LLFIN )LLFIN = LSTCON                             00920000
      DO 70 K =1,LSTCON                                                 00930000
      DO 50 L =1,NRISM                                                  00950000
      SUM = 0.                                                          00960000
      J2 = 0                                                            00970000
      DO 30 I=1,NBLKS                                                   00990000
       IF(KLSBLK(I).LE.0) GO TO 30                                      01000000
      M1  = MOST - INDNOS(I) + INDLOW                                   01050000
      J1 = J2 + 1                                                       01060000
      J2 = J2 + KLSBLK(I)                                               01070000
      DO 28 J=J1,J2                                                     01110000
      SUM = SUM + STRESS(L,J) * X(M1,K)                                 01120000
      M1 = M1-1                                                         01130000
   28 CONTINUE                                                          01140000
   30 CONTINUE                                                          01160000
      FSS(K,L) = SUM                                                    01170000
      IF(SUM.GT.OPMIN(L))GO TO 35                                       01210000
      OPMIN(L) = SUM                                                    01220000
      KONMIN(L) = K                                                     01230000
   35 IF(SUM.LT.OPMAX(L)) GO TO 50                                      01250000
      OPMAX(L) = SUM                                                    01260000
      KONMAX(L) = K                                                     01270000
   50 CONTINUE                                                          01290000
   70 CONTINUE                                                          01310000
      WRITE(13) NUMREC,KIND ,LSTCON,NRISM,((FSS(K,L),K=1,LSTCON),L=1,NRI01320000
     1SM),NODEAR,KONMAX,KONMIN,OPMAX,OPMIN,LLIN,LLFIN,KMAX,KMIN,VOLUME  01330000
      IF(MRECS.GT.NSRECS) GO TO 5                                       01340000
      END FILE 13                                                       01350000
      REWIND 13                                                         01360000
      RETURN                                                            01370000
      END                                                               01380000
```

MEMBER NAME MAIN685

```
      COMMON NELRLC, MRECS, NREC11, LSTCON, NBAND, MXBAND,              00110000
     1        MXCLND, MXNODS, NBASE, MSTIFF, MSTRES, LIN,               00120000
     2        NSFREC, NOGO, NSTRPS, KNTLDS, MUDDY1, MUDDY2              00130000
      REWIND 10                                                         00170000
      REWIND 12                                                         00210000
      REWIND 9                                                          00250000
      REWIND 4                                                          00290000
      CALL DEFL                                                         00330000
      REWIND 11
      CALL FSTR
      CALL EXIT
      STOP 7                                                            00400000
      END                                                               00410000
```

MEMBER NAME UNPK685

```
      SUBROUTINE UNPAK(NUMPKD,IUNPKD,KSAVE)                             00010000
      DIMENSION IUNPKD(6)                                               00080000
      KSAVE = 0                                                         00090000
      K = NUMPKD                                                        00100000
      DO 15 L=1,6                                                       00110000
      I = 6 - L                                                         00120000
      NED = 10 ** I                                                     00130000
      IUNPKD(L) = K / NED                                               00140000
      IF( IUNPKD(L).EQ.0) KSAVE= KSAVE+ 1                               00150000
      K = K - NED * IUNPKD(L)                                           00160000
   15 CONTINUE                                                          00170000
      RETURN                                                            00180000
      END
```

```
MEMBER NAME  A3LNK689
      SUBROUTINE A3LNK1(MS)
      IMPLICIT REAL*8 (A-G,O-Z)
      REAL*8 HMS
      COMMON/LINKA1/NTOTAL,NORET,NTAPE
      COMMON /LINKA/ LAP(150),
     1         XNONOD,XNOSAT,XNORET,XNORED,XNOPOP,AXNOD,EE(3),XMU(3),
     2 XNMASV,XNEIG,XNMASG,SCALE,EVSAVE,XKSAVE,XNSPEC,
     3 AS,FSM,RSM,DM,UH,WONE,DINV,FDEFL,FLOAD,OMEGA
      DIMENSION WORDS(4),NODE1(2750),NODE2(2750),TEMP1(2750),
     1 TEMP2(2750),TEMP3(2750),TEMP(4),
     2 X(600),Y(600),Z(600),NONE(600),NTWO(600),
     3MAP(1850),KK(6),DUM(4),
     4 XBK(12,12),ROW(1850),ROW2(12)
      DIMENSION TWOR(4),SEMP(4)
      DIMENSION XMS(3000),HMS(6)
      DIMENSION XX(12)
      DIMENSION XIP(12)
      DIMENSION XCON(2750)
      DIMENSION VOLUM(2750)
      DIMENSION OPTION(2),DATA(30),IOP(4)
      DATA OPTION/'YES','NO'/
      EQUIVALENCE(DATA(1),XNONOD)
      DATA TWOR/4H    ,4H    ,4H    ,4H    /
      VOA=0.
      VOB=0.
      VOS=0.
      NSPEC=XNSPEC
      NONOD=XNONOD
      NOSAT=XNOSAT
      NORET=XNORET
      NORED=XNORED
      NOPROP=XNOPOP
      MAXNOD=AXNOD
      NMASV=XNMASV
      NEIG=XNEIG
      NMASG=XNMASG
      NTOTAL=NORET+NORED
      NN=NTOTAL
      NODEL=6*NONOD-NORET-NORED
      DO 4629  I=1,4
 4629 IOP(I)=DATA(I+19)
      NOYE=2
      IF(UH.EQ.1) NOYE=1
      CALL LOADER (SKP,2)
      WRITE(6,904) MS
  904 FORMAT(20X,'**** MODE SHAPE NO. ',I2,' ****',/)
  999 FORMAT(1H0,29X,'ENERGY ANALYSIS PROGRAM'/22X,
     1    '        C52   SCIARRA / SHATZ'77)
      WRITE (6,999)
      WRITE (6,985)NONOD,NOSAT,NORET,NORED,NOPROP,MAXNOD,NODEL,NTOTAL
     1            ,NMASV,NEIG,NMASG,SCALE
  985 FORMAT (9X,15H NO. OF NODES = I6,/ 9X,25H NO. OF STRUCTURAL ELEMEN
     1         4HTS = I6 / 9X,23H NO. OF RETAINED DOFS = I6 / 9X,
     2         22H NO. OF REDUCED DOFS = I6 / 9X,16H NO. OF MATERIAL
     3         13H PROPERTIES = I6/ 9X,21H MAX. NODE NO. USED = I6 / 9X,
     4         22H NO. OF DELETED DOFS = I6 / 9X,20H FULL SIZE OF MATRIX
     5         19H BEFORE REDUCTION = I6 / 9X,18H NO. OF VARIATION=I6/ 9X
     6         ,21H NO. OF EIGENVECTORS=I6/ 9X,20H NO. OF MASS GROUPS=I6
     7 / 9X ' SCALE FACTOR=',E15.6 )
      WRITE(6,4915) NSPEC
 4915 FORMAT( 10X 'NO. OF SPECIAL ELEMENTS=',I6 )
      WRITE(6,4916) FLOAD,FDEFL,OMEGA
 4916 FORMAT( 10X 'NO. OF LOAD CASES=',F6.0 / 10X 'NO. OF DEFLECTIONS=',
     1 F6.0 / 10X 'EXCITING FREQ(CPS)=',F6.0 )
      WRITE(6,986) (EE(I),XMU(I),I=1,NOPROP)
  986 FORMAT( / 18X 20H MATERIAL PROPERTIES /10X 8HMODULUS 19X 7HPOISSON
     1 6H RATIO (/ 6X F15.2, 15X F10.4))
      WRITE(6,9676) (OPTION(IOP(I)+1),I=1,4),OPTION(NOYE)
 9676 FORMAT( / 15X 'INPUT OPTIONS' / 10X 'INPUT', 14X A4 / 10X 'FULL K'
     1 , 13X A4 / 10X 'FINAL K', 12X A4 / 10X 'DYNAMIC MATRIX', 5X A4 /
     2 10X 'UPPER HESSENBERG', 3X A4 / )
      WRITE(6,9677) WONE
```

```
 9677 FORMAT ( 10X 'EIGENVECTORS CALCULATED FOR EIGENVALUES GREATER THAN00576000
     1', E13.6 )                                                        00577000
      IF(AS.EQ.1.) GO TO 686                                            00611000
      ICOUNT=1                                                          00620000
      KOUNT=0                                                           00630000
      CALL LOADER (SKP,2)                                               00640000
      WRITE(6,904) MS
  987 FORMAT (32X,14H ACTION TABLE /1X 80(1H-)/                         00690000
     1 3X 'NODE', 7X 'X', 11X 'Y', 9X 'Z', 6X 'NONE', 3X 'NTWO' / )     00700000
      WRITE (6,987)                                                     00720000
  686 CONTINUE                                                          00720100
      JJ=1                                                              00760000
      DO 1 I =1,NONOD                                                   00770000
      READ (5,800) NODE,X(NODE),Y(NODE),Z(NODE),NONE(NODE),NTWO(NODE),  00780000
     1              (DUM(IJ1),IJ1=1,4)                                  00790000
      IF(AS.EQ.1.) GO TO 687                                            00791000
      IF(ICOUNT.NE.51) GO TO 9                                          00800000
      CALL LOADER (SKP,2)                                               00810000
      WRITE(6,904) MS
      WRITE (6,987)                                                     00820000
      ICOUNT=0                                                          00830000
    9 ICOUNT=ICOUNT+1                                                   00840000
  800 FORMAT (I3,3E10.0,2I3,4A 6)                                       00850000
      WRITE(6,998)NODE,I,X(NODE),Y(NODE),Z(NODE),NONE(NODE),NTWO(NODE),
     1              (DUM(IJ1),IJ1=1,4)                                  00870000
  998 FORMAT(1H ,I3,I4,2X,3F11.4,3X,I3,3X,I3,2X,4A6)
  687 CALL UNPACK(NONE(NODE),NTWO(NODE),KK)                             00920000
      DO 700 L =1,6                                                     00940000
      MAP(JJ)=NODE                                                      00970000
      JJ=JJ+1                                                           00980000
  700 CONTINUE                                                          01040000
    1 CONTINUE                                                          01050000
      DO 50 I=1,3000                                                    01050200
   50 XMS(I)=0.                                                         01050220
   30 READ(11,END=90)NODE,LL,(HMS(I),I=1,6)                             01050240
      IF(LL.NE.MS)GO TO 30                                              01050242
      M=6*NODE                                                          01050246
      DO 70 K=1,3                                                       01050249
      XMS(M-3+K)=HMS(K)                                                 01050250
   70 XMS(M-6+K)=HMS(K+3)                                               01050252
      GO TO 30                                                          01050270
   90 CONTINUE                                                          01050272
      CALL LOADER (SKP,2)                                               01200000
      WRITE(6,904) MS
  989 FORMAT(1X 80(1H-)/ 18X,'MAP'// (3X,16I4))                         01230000
      NOW=NONOD*6                                                       01230100
      WRITE(6,989)(MAP(I),I=1,NOW)                                      01240000
  911 FORMAT (1X  80(1H-))                                              01250000
      WRITE (6,911)                                                     01260000
  150 IF(AS.EQ.1.) GO TO 688                                            01311000
      CALL LOADER(FLIP,2)                                               01320000
      WRITE(6,904) MS
  688 NAXIAL=0                                                          01340000
      NSKIN=0                                                           01350000
      NBEAM=0                                                           01360000
 6066 NAP=NOSAT+NSPEC                                                   01384000
      I=0                                                               01384100
      DO 11  INDEX=1,NAP                                                01385000
      IF(INDEX.GT.NSPEC) GO TO 9065                                     01386000
      DO 28  LS=1,4                                                     01386100
   28 WORDS(LS)=TWOR(LS)                                                01386200
      NM=0                                                              01387000
      MO=0                                                              01388000
      READ(5,5055) NTYP,LN1,LN2,LN3,LN4,MC,(SEMP(L),L=1,4)              01389000
 5055 FORMAT(5I3,I1,4E10.0)                                             01389010
      WRITE(6,2371) NTYP, LN1,LN2,LN3,LN4,MC,(SEMP(L),L=1,4)            01389020
 2371 FORMAT( 1X 80(1H-) / 21X 'SPECIAL ELEMENT' / '  TYPE =',I4,       01389030
     1 '  LN1 =',I4, '  LN2 =',I4, '  LN3 =', I4, '  LN4 =', I4 /       01389040
     2 '  PROPERTIES =',I2, 4( ',', E14.5))                             01389050
      IF(NTYP.NE.5) GO TO 3035                                          01389100
```

```
  781 N1=LN1
      N2=LN2
      N3=LN3
      IF(NM.EQ.1) GO TO 5631
      NTYPE=2
      TEMP(1)=SEMP(4)
      TEMP(2)=0.
      TEMP(3)=0.
      NM=1
      GO TO 7777
 5631 NTYPE=4
      DO 6181  J=1,3
 6181 TEMP(J)=SEMP(J)
      NM=0
      GO TO 7777
 3035 IF(NTYP.NE.6) GO TO 4045
  881 IF(MO.GT.2) GO TO 3611
      NTYPE=3
      N3=LN4
      DO 6212  J=1,3
 6212 TEMP(J)=SEMP(1)
      IF(MO.EQ.2) GO TO 3711
      N1=LN1
      N2=LN2
      MO=2
      GO TO 7777
 3711 N1=LN2
      N2=LN3
      MO=3
      GO TO 7777
 3611 NTYPE=2
      N3=0
      TEMP(2)=0.
      TEMP(3)=0.
      IF(MO.EQ.4) GO TO 3811
      N1=LN1
      N2=LN2
      TEMP(1)=SEMP(2)
      MO=4
      GO TO 7777
 3811 N1=LN3
      N2=LN4
      TEMP(1)=SEMP(3)
      MO=0
      GO TO 7777
 4045 WRITE(6,7227)
 7227 FORMAT( // 5X 'INCORRECT IDENTIFICATION FOR SPECIAL ELEMENT' )
      CALL EXIT
 9065 READ(5,801) NTYPE,N1,N2,N3,MC,(TEMP(J),J=1,3),(WORDS(J2),J2=1,4)
  801 FORMAT (2X,I1,3I3,I1,   3E10.0,4A6)
      GO TO 8777
 7777 NOSAT=NOSAT+1
 8777 I=I+1
      IF(NTYPE.EQ.0) NTYPE=3
      IF (MC.EQ.0)  MC=1
      IF ((NTYPE.NE.3).OR .(TEMP(2).NE.0.)) GO TO 100
      TEMP(2)=TEMP(1)
      TEMP(3)=TEMP(1)
  100 CONTINUE
      IF (NTYPE.NE.2) GO TO 101
      IF(AS.EQ.1.) GO TO 681
      WRITE (6,992) N1,N2,MC,TEMP(1),(WORDS(J),J=1,4)
  992 FORMAT (1X80(1H-)/21X14H AXIAL ELEMENT/2X3HN1=I4,2X3HN2=I4,2X,
     1          14HMATERIAL CODE=I2,2X, 5HAREA=E12.5,2X,4A6 )
  681 NAXIAL=NAXIAL+1
      NODE1(I)=N1*100+10*NTYPE+MC
      NODE2(I)=N2*1000+N3
      TEMP1(I)=TEMP(1)
      TEMP2(I)=TEMP(2)
      TEMP3(I)=TEMP(3)
```

```
      GO TO 10                                                          01750000
  101 IF (NTYPE.NE.3) GO TO 102                                         01770000
      IF(AS.EQ.1.) GO TO 682                                            01801000
      WRITE (6,993)                                                     01810000
  993 FORMAT (1X 80(1H-)/24X 25H TRIANGULAR SKIN ELEMENT  )             01820000
      WRITE (6,994) N1,N2,N3,MC,(TEMP(J),J=1,3),(WORDS(J),J=1,4)        01830000
  994 FORMAT ( 2X, 3HN1=I4, 2X,                                         01840000
     1        3HN2=I4,2X,3HN3=I4,2X,14HMATERIAL CODE=I2,2X,3HTS=F7.4,2X,01850000
     2      3HTX=F7.4/2X 3HTY=F7.4, 2X 4A6 )                            01860000
  682 NSKIN=NSKIN+1                                                     01870000
      NODE1(I)=N1*100+10*NTYPE+MC                                       01890000
      NODE2(I)=N2*1000+N3                                               01900000
      TEMP1(I)=TEMP(1)                                                  01910000
      TEMP2(I)=TEMP(2)                                                  01920000
      TEMP3(I)=TEMP(3)                                                  01930000
      GO TO 10                                                          01940000
  102 IF (NTYPE.NE.4) GO TO 103                                         01960000
      IF(AS.EQ.1.) GO TO 683                                            01991000
      WRITE (6,995)                                                     02000000
  995 FORMAT (1X 80(1H-)/ 21X 14H BEAM ELEMENT )                        02010000
  954 FORMAT ( 2X, 3HN1=I4, 2X,                                         02020000
     1        3HN2=I4,2X,3HN3=I4,2X,14HMATERIAL CODE=I2,2X,3HIY=E12.5,2X02030000
     2      , 3HJX=E12.5/2X3HAW=E12.5, 2X 4A6 )                         02040000
      WRITE (6,954) N1,N2,N3,MC,(TEMP(J),J=1,3),(WORDS(L),L=1,4)        02050000
  683 NBEAM=NBEAM+1                                                     02060000
      NODE1(I)=N1*100+10*NTYPE+MC                                       02070000
      NODE2(I)=N2*1000+N3                                               02080000
      TEMP1(I)=TEMP(1)                                                  02090000
      TEMP2(I)=TEMP(2)                                                  02100000
      TEMP3(I)=TEMP(3)                                                  02110000
  103 CONTINUE                                                          02170000
   10 IF(INDEX.GT.NSPEC) GO TO 11                                       02180000
      IF(NM.EQ.1) GO TO 781                                             02181000
      IF(MO.NE.0) GO TO 881                                             02182000
   11 CONTINUE                                                          02183000
      WRITE (6,911)                                                     02190000
      NTOT=NAXIAL+NSKIN+NBEAM                                           02200000
      WRITE (6,977) NTOT,NAXIAL,NSKIN,NBEAM                             02300000
  977 FORMAT (9X,37H TOTAL NUMBER OF STRUCTURAL ELEMENTS I6/9X,         02310000
     1              23H NO. OF AXIAL ELEMENTS I6 /9X,                   02320000
     2              23H NO. OF SKIN ELEMENTS  I6/9X,                    02330000
     3              23H NO. OF BEAM ELEMENTS  I6/)                      02340000
      CALL LOADER (SKP,2)                                               02510100
      WRITE(6,904) MS
      DO 42 J =1,NOSAT                                                  02520000
      N1=NODE1(J)/100                                                   02540000
      N2=NODE2(J)/1000                                                  02550000
      N3=NODE2(J)-1000*N2                                               02560000
      NO1=N1                                                            02570000
      NTYPE=(NODE1(J)-100*NO1)/10                                       02590000
   45 MC=NODE1(J)-100*NO1-10*NTYPE                                      02760000
      E=EL(MC)                                                          02770000
      TEMP(1)=XMU(MC)                                                   02780000
      IF (NTYPE.EQ.2) TEMP(1)=TEMP1(J)                                  02790000
      TEMP(2)=TEMP1(J)                                                  02800000
      TEMP(3)=TEMP2(J)                                                  02810000
      TEMP(4)=TEMP3(J)                                                  02820000
      DO 76 N=1,12                                                      02828001
   76 XX(N)=0.                                                          02828010
      DO 72 M=1,NOW                                                     02828020
      IF(MAP(M).EQ.N1)GO TO 74                                          02828030
   72 CONTINUE                                                          02828040
   74 DO 73 N=1,3                                                       02828050
      XX(N)=XMS(M+N-1)                                                  02828060
      IF(NTYPE.EQ.4)XX(N+3)=XMS(M+N+2)                                  02828070
   73 CONTINUE                                                          02828080
      DO 89 M=1,NOW                                                     02828090
      IF(MAP(M).EQ.N2)GO TO 81                                          02829010
   89 CONTINUE                                                          02829020
   81 DO 82 N=4,6                                                       02829030
```

```
      IF(NTYPE.EQ.4)GO TO 77                                            02829031
      XX(N)=XMS(M+N-4)                                                  02829033
      GO TO 82                                                          02829035
   77 XX(N+3)=XMS(M+N-4)                                                02829037
      XX(N+6)=XMS(M+N-1)                                                02829039
   82 CONTINUE                                                          02829040
      IF(NTYPE.NE.3)GO TO 84                                            02829041
      DO 86 M=1,NOW                                                     02829043
      IF(MAP(M).EQ.N3)GO TO 85                                          02829045
   86 CONTINUE                                                          02829047
   85 DO 87 N=7,9                                                       02829049
   87 XX(N)=XMS(M+N-7)                                                  02829050
   84 CONTINUE                                                          02829052
   79 FORMAT(1H0,3E14.7)                                                02829056
  900 FORMAT(1H0,7I8)                                                   02860200
      CALL SETUP(NTYPE,N1,N2,N3,E,TEMP,X,Y,Z,NONE,NTWO,XBK,IZ,VO)       02870000
      IF (NTYPE.EQ.4) CALL RESORT (XBK)                                 02880000
   94 FORMAT(1H0,6E14.7)                                                02880200
      DO 96 K=1,12                                                      02880300
      XIP(K)=0.                                                         02880400
      DO 96 I=1,12                                                      02880500
   96 XIP(K)=XX(I)*XBK(I,K)+XIP(K)                                      02880510
      XIP(1)=XIP(1)*XX(1)                                               02880520
      DO 97 I=2,12                                                      02880530
   97 XIP(I)=XIP(I)*XX(I)+XIP(I-1)                                      02880540
      XCON(J)=XIP(12)                                                   02880541
      IF(NTYPE.NE.2)GO TO 300                                           02880542
      WRITE(6,992)N1,N2,MC,TEMP(1)                                      02880544
      VOA=VO+VOA                                                        02880546
      VOLUM(J)=VO                                                       02880547
      GO TO 302                                                         02880549
  300 IF(NTYPE.NE.4)GO TO 301                                           02880550
      WRITE(6,995)                                                      02880552
      WRITE(6,994)N1,N2,N3,MC,(TEMP(N),N=2,4)                           02880553
      VOB=VO+VOB                                                        02880555
      VOLUM(J)=VO                                                       02880557
      GO TO 302                                                         02880559
  301 WRITE(6,993)                                                      02880560
      WRITE(6,994)N1,N2,N3,MC,(TEMP(N),N=2,4)                           02880561
      VOS=VO+VOS                                                        02880562
      VOLUM(J)=VO                                                       02880565
  302 CONTINUE                                                          02880570
      WRITE(6,901)XIP(12),VO                                            02880572
  901 FORMAT(1H ,10X,'STRAIN ENERGY = ',E15.7,3X,'VOLUME = ',E15.7)     02880573
  902 FORMAT(1H ,10X,'TOTAL VOLUME = ',E15.7/11X,
     1'TOTAL AXIAL VOLUME = ',E15.7/11X,                                02880575
     2'TOTAL BEAM VOLUME = ',E15.7/11X,'TOTAL SKIN VOLUME = ',E15.7)    02880576
   42 CONTINUE                                                          03250000
      REWIND 8
      TSE=0.0
      DO 17 N=1,NOSAT
      TSE=TSE+XCON(N)
      STD=XCON(N)/VOLUM(N)
      WRITE(8) STD,NODE1(N),NODE2(N),TEMP1(N),TEMP2(N),TEMP3(N)
   17 CONTINUE
      WRITE(6,903) TSE
  903 FORMAT(//1H ,10X,'TOTAL STRAIN ENERGY = ',E15.7)
      S=0
      N=0
      WRITE(8) S,N,N,S,S,S
      MOSAT=NOSAT-1                                                     03250010
      DO 12 N=1,MOSAT                                                   03250100
      IP1=N+1                                                           03250110
      DO 12 K=IP1,NOSAT                                                 03250120
      IF(XCON(N).LE.XCON(K))GO TO 12                                    03250130
      SORT=XCON(N)                                                      03250140
      XCON(N)=XCON(K)                                                   03250150
      XCON(K)=SORT                                                      03250152
      SORT=VOLUM(N)                                                     03250154
      VOLUM(N)=VOLUM(K)                                                 03250155
      VOLUM(K)=SORT                                                     03250156
```

```
      NORT=NODE1(N)                                                     03250161
      NODE1(N)=NODE1(K)                                                 03250162
      NODE1(K)=NORT                                                     03250163
      NORT=NODE2(N)                                                     03250164
      NODE2(N)=NODE2(K)                                                 03250165
      NODE2(K)=NORT                                                     03250166
      SORT=TEMP1(N)                                                     03250170
      TEMP1(N)=TEMP1(K)                                                 03250171
      TEMP1(K)=SORT                                                     03250172
      SORT=TEMP2(N)                                                     03250173
      TEMP2(N)=TEMP2(K)                                                 03250174
      TEMP2(K)=SORT                                                     03250175
      SORT=TEMP3(N)                                                     03250176
      TEMP3(N)=TEMP3(K)                                                 03250177
      TEMP3(K)=SORT                                                     03250178
   12 CONTINUE                                                          03250179
      REWIND 9
      WRITE(9) TVOL,VOA,VOB,VOS,NOSAT,MS
      WRITE(9)  (XCON(I),I=1,NOSAT)
      WRITE(9) (VOLUM(I),I=1,NOSAT)
      WRITE(9) (NODE1(I),I=1,NOSAT)
      WRITE(9) (NODE2(I),I=1,NOSAT)
      WRITE(9) (TEMP1(I),I=1,NOSAT)
      WRITE(9) (TEMP2(I),I=1,NOSAT)
      WRITE(9) (TEMP3(I),I=1,NOSAT)
      RETURN
      ENTRY A3LNK2
      REWIND 8
      REWIND 9
      READ(9) TVOL,VOA,VOB,VOS,NOSAT,MS
      READ(9)  (XCON(I),I=1,NOSAT)
      READ(9) (VOLUM(I),I=1,NOSAT)
      READ(9) (NODE1(I),I=1,NOSAT)
      READ(9) (NODE2(I),I=1,NOSAT)
      READ(9) (TEMP1(I),I=1,NOSAT)
      READ(9) (TEMP2(I),I=1,NOSAT)
      READ(9) (TEMP3(I),I=1,NOSAT)
      TVOL=VOA+VOS                                                      03250180
      WRITE(6,902)TVOL,VOA,VOB,VOS                                      03250181
      DO 13 ML=1,2                                                      03250182
      CALL LOADER (SKP,2)                                               03250184
      WRITE(6,904) MS
      DO 13 NL=1,NOSAT                                                  03250188
      N=NOSAT-NL+1                                                      03250190
      IF(ML.EQ.2) READ(8)    STD,NODE1(N),NODE2(N),TEMP1(N),TEMP2(N),
     1                       TEMP3(N)
      N1=NODE1(N)/100                                                   03250210
      N2=NODE2(N)/1000                                                  03250220
      N3=NODE2(N)-1000*N2                                               03250240
      NO1=N1                                                            03250250
      NTYPE=(NODE1(M)-100*NO1)/10                                       03250260
      MC=NODE1(N)-100*NO1-10*NTYPE                                      03250270
      E=EF(MC)                                                          03250280
      TEMP(1)=XMU(MC)                                                   03250290
      IF(NTYPE.EQ.2)TEMP(1)=TEMP1(N)                                    03250300
      TEMP(2)=TEMP1(N)                                                  03250310
      TEMP(3)=TEMP2(N)                                                  03250320
      TEMP(4)=TEMP3(N)                                                  03250330
      IF(NTYPE.NE.2)GO TO 14                                            03250340
      WRITE(6,992)N1,N2,MC,TEMP(1)                                      03250350
      IF(ML.EQ.2)WRITE(6,1003)STD                                       03250352
 1003 FORMAT(1H ,25X,'STRAIN DENSITY = ',E14.7)                         03250353
      IF(ML.EQ.2)GO TO 15                                               03250354
      VOAT=VOLUM(N)/TVOL                                                03250355
      VOAA=VOLUM(N)/VOA                                                 03250356
      WRITE(6,1002)XCON(N),VOAA,VOAT                                    03250357
 1002 FORMAT(1H ,25X,'STRAIN ENERGY = ',E15.7,/3X,                      03250358
     1'EL. VOL. / GROUP VOL.=',E15.7,2X,'EL. VOL. / TOTAL VOL.= ',E15.7)03250359
      GO TO 15                                                          03250360
   14 IF(NTYPE.NE.4)GO TO 16                                            03250370
```

```
      WRITE(6,995)                                                      03250372
      WRITE(6,994)N1,N2,N3,MC,(TEMP(K),K=2,4)                           03250380
      IF(ML.EQ.2)WRITE(6,1004)STD                                       03250381
      IF(ML.EQ.2)GO TO 15                                               03250383
      VOAT=VOLUM(N)/TVOL                                                03250385
      VOAB=VOLUM(N)/VOB                                                 03250387
      WRITE(6,1005)XCON(N),VOAB,VOAT                                    03250388
      GO TO 15                                                          03250390
   16 WRITE(6,993)                                                      03250392
      WRITE(6,994)N1,N2,N3,MC,(TEMP(K),K=2,4)                           03250400
      IF(ML.EQ.2)WRITE(6,1004)STD                                       03250401
 1004 FORMAT(1H+,25X,'STRAIN DENSITY = ',E14.7)                         03250402
      IF(ML.EQ.2)GO TO 15                                               03250403
      VOAT=VOLUM(N)/TVOL                                                03250404
      VOAS=VOLUM(N)/VOS                                                 03250405
      WRITE(6,1005)XCON(N),VOAS,VOAT                                    03250406
 1005 FORMAT(1H+,25X,'STRAIN ENERGY = ',E15.7,/3X,                      03250407
     1'EL. VOL. / GROUP VOL.=',E15.7,2X,'EL. VOL. / TOTAL VOL.= ',E15.7)03250408
   15 CONTINUE                                                          03250410
   13 CONTINUE                                                          03250430
  500 RETURN                                                            03710000
      END                                                               03720000


MEMBER NAME  BEAM689
      SUBROUTINE BEAM(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,E,XMU,YI,XJ,AW,BMAT,VO)00020000
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION BMAT(12,12),OME(6,12),BK(6,6),TEMP(6,12),D(3,3)         00030000
      DOUBLE PRECISION L1,I1,J1,MU
      SQRT(X) = DSQRT(X)
      ABS(X) = DABS(X)
      MU=XMU
      X31=X3-X1                                                         00070000
      Y31=Y3-Y1                                                         00080000
      Z31=Z3-Z1                                                         00100000
      X12=X1-X2                                                         00120000
      Y12=Y1-Y2                                                         00170000
      Z12=Z1-Z2                                                         00180000
      X13=X1-X3                                                         00190000
      Y13=Y1-Y3                                                         00200000
      Z13=Z1-Z3                                                         00210000
      SU1=(1./3.)                                                       00220000
      SU3=(2./3.)                                                       00230000
      I1=YI                                                             00240000
      A=AW                                                              00260000
      J1=XJ                                                             00270000
      SL=SQRT(X12**2+Y12**2+Z12**2)                                     00280000
      VO=A*SL                                                           00300000
      G=E/(2.*(1.+MU))                                                  00300010
      SA=0.                                                             00310000
      IF(A.EQ.0.) GO TO 105                                             00311000
      SA=(12.*E*I1)/(G*A*SL**2)                                         00312000
  105 F=(6.*E*I1)/(SL**3*(1.+SA))                                       00320000
      DO 5 I =1,6                                                       00330000
      DO 6 J =1,6                                                       00380000
    6 BK(I,J)=0.0                                                       00390000
    5 CONTINUE                                                          00400000
      BK(1,1)=2.*F                                                      00410000
      BK(3,1)=-SL*F                                                     00420000
      BK(4,1)=-2.*F                                                     00430000
      BK(6,1)=-SL*F                                                     00440000
      BK(2,2)=G*J1/SL                                                   00450000
      BK(5,2)=-(G*J1)/SL                                                00470000
      BK(3,3)=SU3*SL**2*F*(1.+SA/4.)                                    00480000
      BK(4,3)=SL*F                                                      00500000
      BK(6,3)=SU1*SL**2*F*(1.-SA/2.)                                    00510000
      BK(4,4)=2.*F                                                      00520000
      BK(6,4)=SL*F                                                      00540000
      BK(5,5)=G*J1/SL                                                   00550000
                                                                        00570000
```

```
      BK(6,6)=SU3*SL**2*F*(1.+SA/4.)                                  00590000
      DO  8 I =1,6                                                    00610000
      DO  8 J =1,6                                                    00620000
    8 BK(I,J)=BK(J,I)                                                 00630000
      X21=X2-X1                                                       00690000
      Y21=Y2-Y1                                                       00700000
      Z21=Z2-Z1                                                       00710000
      DX=SQRT(X21**2+Y21**2+Z21**2)                                   00730000
      XBX=X21/DX                                                      00740000
      YBX=Y21/DX                                                      00750000
      ZBX=Z21/DX                                                      00760000
      DO 100 J =1,3                                                   00780000
  100 D(J,1)=1.                                                       00790000
      D(1,2)=Y1                                                       00810000
      D(2,2)=Y2                                                       00820000
      D(3,2)=Y3                                                       00830000
      D(2,3)=Z2                                                       00850000
      D(1,3)=Z1                                                       00860000
      D(3,3)=Z3                                                       00870000
      CALL DET (D,A)                                                  00890000
      DO 101 J =1,3                                                   00910000
  101 D(J,2)=1.                                                       00920000
      D(1,1)=X1                                                       00940000
      D(2,1)=X2                                                       00950000
      D(3,1)=X3                                                       00960000
      CALL DET (D,B)                                                  00980000
      DO 102 J =1,3                                                   01000000
  102 D(J,3)=1.                                                       01010000
      D(1,2)=Y1                                                       01030000
      D(2,2)=Y2                                                       01040000
      D(3,2)=Y3                                                       01050000
      CALL DET (D,C)                                                  01070000
      BD=SQRT(A**2+B**2+C**2)                                         01090000
      XBY=A/BD                                                        01100000
      YBY=B/BD                                                        01110000
      ZBY=C/BD                                                        01120000
      XBZ=(YBX*ZBY)-(YBY*ZBX)                                         01140000
      YBZ=(XBY*ZBX)-(XBX*ZBY)                                         01150000
      ZBZ=(XBX*YBY)-(XBY*YBX)                                         01160000
      DO 9 I =1,6                                                     01180000
      DO 10 J=1,12                                                    01190000
   10 OME(I,J)=0.0                                                    01200000
    9 CONTINUE                                                        01210000
      OME(1,1)=XBZ                                                    01230000
      OME(1,2)=YBZ                                                    01240000
      OME(1,3)=ZBZ                                                    01250000
      OME(4,4)=XBZ                                                    01270000
      OME(4,5)=YBZ                                                    01280000
      OME(4,6)=ZBZ                                                    01290000
      OME(2,7)=XBX                                                    01310000
      OME(3,7)=XBY                                                    01320000
      OME(2,8)=YBX                                                    01330000
      OME(3,8)=YBY                                                    01340000
      OME(2,9)=ZBX                                                    01350000
      OME(3,9)=ZBY                                                    01360000
      OME(5,10)=XBX                                                   01380000
      OME(5,11)=YBX                                                   01390000
      OME(5,12)=ZBX                                                   01400000
      OME(6,10)=XBY                                                   01420000
      OME(6,11)=YBY                                                   01430000
      OME(6,12)=ZBY                                                   01440000
      DO 3 I =1,6                                                     01470000
      DO 3 J =1,12                                                    01480000
      TEMP(I,J)=0.0                                                   01490000
      DO 3 K =1,6                                                     01510000
    3 TEMP(I,J)=TEMP(I,J)+BK(I,K)*OME(K,J)                            01520000
      DO 11 I =1,12                                                   01540000
```

```
      DO 11 J =1,12                                                     01550000
      BMAT(I,J)=0.0                                                     01560000
      DO 13 K =1,6                                                      01570000
   13 BMAT(I,J)=BMAT(I,J)+OME(K,I)*TEMP(K,J)                            01590000
      IF (ABS(BMAT(I,J)).GT..1) GO TO 11                                01600000
      BMAT(I,J)=0.0                                                     01610000
   11 CONTINUE                                                          01620000
      RETURN                                                            01640000
      END                                                               01650000
```

```
MEMBER NAME  DET689
      SUBROUTINE DET (D,ANS)                                            00020000
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION D(3,3)                                                  00040000
      ANS=D(1,1)*D(2,2)*D(3,3)+D(1,2)*D(2,3)*D(3,1)+D(1,3)*D(2,1)*D(3,2)00060000
      ANS=ANS-(D(3,1)*D(2,2)*D(1,3)+D(3,2)*D(2,3)*D(1,1)+D(3,3)*D(2,1)* 00070000
     1        D(1,2))                                                   00080000
      RETURN                                                            00100000
      END                                                               00110000
```

```
MEMBER NAME  LOAD689
      SUBROUTINE LOADER(ARRAY,MODE)
      IMPLICIT REAL*8 (A-H,O-S,U-Z)
      DIMENSION CARD(5),TITLE(19),ARRAY(1)
      GO TO(100,200),MODE
  100 READ(5,101)  NSUB,NOPC,CARD
  101 FORMAT(I4,I1,5E14.7)
      IF(NSUB.LE.0) GO TO 303
      IF(NOPC.LE.0) GO TO 303
      IF(NOPC-6)109,303,108
  108 IF(NOPC-8)102,111,112
  109 DO 110 JCARD=1,NOPC
      JARRAY=NSUB + JCARD - 1
  110 ARRAY(JARRAY) = CARD(JCARD)
      GO TO 100
  102 READ(5,103) TITLE
  103 FORMAT(I6,17A4,A2)
  104 NPAGE = 0
      GO TO 100
  111 RETURN
  112 STOP
  200 NPAGE = NPAGE + 1
      WRITE(6,201) (TITLE(N),N=2,19),NPAGE
  201 FORMAT(1H1,17A4,A2,6H PAGE ,I3,//)
      RETURN
  303 WRITE(6,304)NSUB,NOPC,CARD
  304 FORMAT(19H INVALID DATA CARD ,I4,I1,5E14.7)
      STOP
      END
```

```
MEMBER NAME  MAIN689
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION MSS(20),DATA(30)
      COMMON/LINKA/LAP(150),XNONOD,XNOSAT,XNORET,XNORED,XNOPOP,AXNOD,
     *EE(3),XMU(3),XNMASV,XNEIG,XNMASG,SCALE,EVSAVE,XKSAVE,XNSPEC,
     *AS,FSM,RSM,DM,UH,WONE,OINV,FDEFL,FLOAD,OMEGA
      COMMON/LINKA1/NTOTAL,NORET,NTAPE
      EQUIVALENCE (DATA(1),XNONOD)
      DO 40 I=1,30
   40 DATA(I)=0.
  500 CALL LOADER(DATA,1)
      KILL=1
      READ(2,1000)NMS
      READ(2,2000)(MSS(I),I=1,NMS)
      DO 600 I=1,NMS
      MS=MSS(I)
      REWIND 11
      CALL A3LNK1(MS)
      CALL SORTSD
      CALL A3LNK2
      REWIND 5
      CALL LOADER(DATA,1)
  600 CONTINUE
 1000 FORMAT(I5)
 2000 FORMAT(16I5)
      END
```

```
MEMBER NAME  RESRT689
      SUBROUTINE RESORT (XBK)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XBK(12,12),EXT(12,12),TEMP(12,12)
      DO 1 I =1,12
      DO 1 J =1,12
    1 EXT(I,J)=0.0
      EXT(1,1)=1.
      EXT(2,2)=1.
      EXT(3,3)=1.
      EXT(4,7)=1.
      EXT(5,8)=1.
      EXT(6,9)=1.
      EXT(7,4)=1.
      EXT(8,5)=1.
      EXT(9,6)=1.
      EXT(10,10)=1.
      EXT(11,11)=1.
      EXT(12,12)=1.
      DO 20 I =1,12
      DO 20 J =1,12
      TEMP(I,J)=0.0
      DO 20 K =1,12
   20 TEMP(I,J)=TEMP(I,J)+EXT(I,K)*XBK(K,J)
      DO 30 I =1,12
      DO 30 J =1,12
      XBK(I,J)=0.0
      DO 30 K =1,12
   30 XBK(I,J)=XBK(I,J)+TEMP(I,K)*EXT(J,K)
      RETURN
      END
```

```
MEMBER NAME  SETUP689
      SUBROUTINE SETUP (NTYPE,N1,N2,N3,E,TEMP,X,Y,Z,NONE,NTWO,XBK,IZ,VO)00010000
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XBK(12,12),X(300 ),Y(300 ),Z(300 ),NONE(4),NTWO(4),TEMP(00040000
     1          4)                                                      00050000
      DO 166 I =1,12                                                    00060000
      DO 166 J =1,12                                                    00070000
  166 XBK(I,J)=0.0                                                      00080000
      IF(NTYPE.NE.2) GO TO 24                                           00110000
      CALL STRING (X(N1),Y(N1),Z(N1),X(N2),Y(N2),Z(N2),E,TEMP(1),XBK,VO)00120000
      IZ=6                                                              00140000
      GO TO 30                                                          00150000
   24 IF(NTYPE.NE.3) GO TO  25                                          00180000
      G=E/(2.*(1.+TEMP(1)))                                             00190000
      CALL SKIN (X(N1),Y(N1),Z(N1),X(N2),Y(N2),Z(N2),X(N3),Y(N3),Z(N3), 00200000
     1          E,G,TEMP(2),TEMP(3),TEMP(4),TEMP(1),XBK,VO)             00210000
      IZ=9                                                              00230000
      GO TO 30                                                          00240000
   25 IF(NTYPE.NE.4) GO TO 26                                           00270000
      CALL BEAM (X(N1),Y(N1),Z(N1),X(N2),Y(N2),Z(N2),X(N3),Y(N3),Z(N3), 00280000
     1          E,TEMP(1),TEMP(2),TEMP(3),TEMP(4),XBK,VO)               00290000
      IZ=12                                                             00300000
      GO TO 30                                                          00310000
   26 WRITE (6,10)                                                      00330000
   10 FORMAT(9X,35HNTYPE NO. IN COLUMN 4 IS NOT 2,3OR4  )               00340000
   30 CONTINUE                                                          00360000
      RETURN                                                            00410000
      END                                                               00420000
```

```
MEMBER NAME  SKIN689
      SUBROUTINE SKIN (X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,E,G,TS,TX,TY,XMU,     00010000
     *TMAT,VO)                                                          00020000
      IMPLICIT REAL*8 (A-H,O-Z)
      DOUBLE PRECISION MU
      SQRT(X)=DSQRT(X)
      ABS(X)=DABS(X)
      DIMENSION TMAT(12,12),D(3,3),OME(9,9),BK(6,6),TEMP(6,9),XX(9),X(6)00050000
      MU=XMU                                                            00100000
      DD=SQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)                         00110000
      COSXBX=(X2-X1)/DD                                                 00120000
      COSYBX=(Y2-Y1)/DD                                                 00130000
      COSZBX=(Z2-Z1)/DD                                                 00140000
      I=1                                                               00160000
      DO 5 J =1,3                                                       00170000
    5 D(J,I)=1.                                                         00180000
      D(1,2)=Y1                                                         00200000
      D(2,2)=Y2                                                         00210000
      D(3,2)=Y3                                                         00220000
      D(1,3)=Z1                                                         00230000
      D(2,3)=Z2                                                         00240000
      D(3,3)=Z3                                                         00250000
      CALL DET (D,A1)                                                   00270000
      D(1,1)=X1                                                         00300000
      D(2,1)=X2                                                         00310000
      D(3,1)=X3                                                         00320000
      I=2                                                               00340000
      DO 6 J = 1,3                                                      00350000
    6 D(J,I)= 1.                                                        00360000
      CALL DET (D,B)                                                    00380000
      I=3                                                               00410000
      DO 7 J =1,3                                                       00420000
    7 D(J,I)=1.                                                         00430000
      D(1,2)=Y1                                                         00450000
      D(2,2)=Y2                                                         00460000
      D(3,2)=Y3                                                         00470000
      CALL DET (D,C)                                                    00490000
      G1=SQRT(A1**2+B**2+C**2)                                          00530000
      COSXBZ=A1/G1                                                      00540000
```

```
      COSYBZ=B/G1                                                        00550000
      COSZBZ=C/G1                                                        00560000
      COSXBY=COSYBZ*COSZBX-COSYBX*COSZBZ                                 00590000
      COSYBY=COSZBZ*COSXBX-COSZBX*COSXBZ                                 00600000
      COSZBY=COSXBZ*COSYBX-COSXBX*COSYBZ                                 00610000
      DO  8 I =1,6                                                       00670000
      DO  9 J =1,9                                                       00680000
    9 OME(I,J)=0.0                                                       00690000
    8 CONTINUE                                                           00700000
      OME(1,1)=COSXBX                                                    00720000
      OME(1,2)=COSYBX                                                    00730000
      OME(1,3)=COSZBX                                                    00740000
      OME(2,1)=COSXBY                                                    00760000
      OME(2,2)=COSYBY                                                    00770000
      OME(2,3)=COSZBY                                                    00780000
      OME(3,4)=COSXBX                                                    00800000
      OME(3,5)=COSYBX                                                    00810000
      OME(3,6)=COSZBX                                                    00820000
      OME(4,4)=COSXBY                                                    00830000
      OME(4,5)=COSYBY                                                    00840000
      OME(4,6)=COSZBY                                                    00850000
      OME(5,7)=COSXBX                                                    00870000
      OME(5,8)=COSYBX                                                    00880000
      OME(5,9)=COSZBX                                                    00890000
      OME(6,7)=COSXBY                                                    00910000
      OME(6,8)=COSYBY                                                    00920000
      OME(6,9)=COSZBY                                                    00930000
      XX(1)=X1                                                           00970000
      XX(2)=Y1                                                           00980000
      XX(3)=Z1                                                           00990000
      XX(4)=X2                                                           01000000
      XX(5)=Y2                                                           01010000
      XX(6)=Z2                                                           01020000
      XX(7)=X3                                                           01030000
      XX(8)=Y3                                                           01040000
      XX(9)=Z3                                                           01050000
      DO 70 I = 1,6                                                      01070000
      X(I)=0.0                                                           01080000
      DO 70 J = 1,9                                                      01090000
   70 X(I)=X(I)+OME(I,J) *XX(J)                                          01100000
      X12=X(1)-X(3)                                                      01110000
      X13=X(1)-X(5)                                                      01120000
      X21=-X12                                                           01130000
      X23=X(3)-X(5)                                                      01140000
      X31=-X13                                                           01150000
      X32=X(5)-X(3)                                                      01160000
      Y12=X(2)-X(4)                                                      01170000
      Y13=X(2)-X(6)                                                      01180000
      Y21=-Y12                                                           01190000
      Y31=-Y13                                                           01200000
      Y23=X(4)-X(6)                                                      01210000
      Y32=-Y23                                                           01220000
      A=.5*(X21*Y31-Y21*X31)                                             01250000
      VO=A*TS                                                            01260000
      GO=TS*G/(4.*A)                                                     01270000
      GX=MU                                                              01290000
      GY=0.                                                              01300000
      IF(TY.EQ.0.) GO TO 105                                             01310000
      GY=MU*TX/TY                                                        01320000
  105 IF(TX.LT.TY) GO TO 50                                              01340000
      GX=0.                                                              01360000
      IF(TX.EQ.0.) GO TO 200                                             01370000
      GX=MU*TY/TX                                                        01380000
  200 GY=MU                                                              01390000
   50 CONTINUE                                                           01410000
      EY=TY*E/(4.*A*(1.-GX*GY))                                          01430000
      EX=TX*E/(4.*A*(1.-GX*GY))                                          01440000
      BK(1,1)=EX*Y23**2+GO*X32**2                                        01460000
      BK(2,1)=Y23*X32*(EY*GY+GO)                                         01470000
      BK(3,1)=EX*Y31*Y23+GO*X13*X32                                      01480000
```

```
      BK(4,1)=EY*GY*X13*Y23+GO*Y31*X32
      BK(5,1)=EX*Y12*Y23+GO*X21*X32
      BK(6,1)=EY*GY*X21*Y23+GO*Y12*X32
      BK(2,2)=EY*X32**2+GO*Y23**2
      BK(3,2)=EX*GX*Y31*X32+GO*X13*Y23
      BK(4,2)=EY*X13*X32+GO*Y31*Y23
      BK(5,2)=EY*GX*Y12*X32+GO*X21*Y23
      BK(6,2)=EY*X21*X32+GO*Y12*Y23
      BK(3,3)=EX*Y31**2+GO*X13**2
      BK(4,3)=EY*GY*X13*Y31+GO*Y31*X13
      BK(5,3)=EX*Y12*Y31+GO*X21*X13
      BK(6,3)=EY*GY*X21*Y31+GO*Y12*X13
      BK(4,4)=EY*X13**2+GO*Y31**2
      BK(5,4)=EX*GX*X13*Y12+GO*X21*Y31
      BK(6,4)=EY*X21*X13+GO*Y12*Y31
      BK(5,5)=EX*Y12**2+GO*X21**2
      BK(6,5)=X21*Y12*(EY*GY+GO)
      BK(6,6)=EY*X21**2+GO*Y12**2
      DO 100 I =1,6
      DO 100 J =1,6
  100 BK(I,J)=BK(J,I)
      DO 10 I =1,6
      DO 10 J =1,9
      TEMP(I,J)=0.0
      DO 10 K =1,6
   10 TEMP(I,J)=TEMP(I,J)+BK(I,K)*OME(K,J)
      DO 11 I =1,9
      DO 11 J =1,9
      TMAT(I,J)=0.0
      DO  3 K =1,6
    3 TMAT(I,J)=TMAT(I,J)+OME(K,I)*TEMP(K,J)
      IF (ABS(TMAT(I,J)).GT..1) GO TO 11
      TMAT(I,J)=0.0
   11 CONTINUE
      RETURN
      END
```

```
MEMBER NAME  SORT689
      SUBROUTINE SORTSD
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION STD(2750),NODE1(2750),NODE2(2750),TEMP1(2750),
     1          TEMP2(2750),TEMP3(2750)
      N=1
      REWIND 8
   10 READ(8) STD(N),NODE1(N),NODE2(N),TEMP1(N),TEMP2(N),TEMP3(N)
      IF(NODE1(N).EQ.0) GO TO 15
      N=N+1
      GO TO 10
   15 N=N-1
   20 M=N-1
      DO 30 L=1,M
      I=L+1
      DO 30 K=I,N
      IF(STD(L).GE.STD(K)) GO TO 30
      SORT=STD(L)
      STD(L)=STD(K)
      STD(K)=SORT
      NORT=NODE1(L)
      NODE1(L)=NODE1(K)
      NODE1(K)=NORT
      NORT=NODE2(L)
      NODE2(L)=NODE2(K)
      NODE2(K)=NORT
      SORT=TEMP1(L)
      TEMP1(L)=TEMP1(K)
      TEMP1(K)=SORT
      SORT=TEMP2(L)
```

```
      TEMP2(L)=TEMP2(K)
      TEMP2(K)=SORT
      SORT=TEMP3(L)
      TEMP3(L)=TEMP3(K)
      TEMP3(K)=SORT
   30 CONTINUE
      REWIND 8
      DO 40 L=1,N
      WRITE(8) STD(L),NODE1(L),NODE2(L),TEMP1(L),TEMP2(L),TEMP3(L)
   40 CONTINUE
      RETURN
      END
```

```
MEMBER NAME  STRNG689
      SUBROUTINE STRING (X1,Y1,Z1,X2,Y2,Z2,E,A,SMAT,VO)
      IMPLICIT REAL*8 (A-H,O-Z)
      DOUBLE PRECISION LAM,MU,NU,L
      DIMENSION SMAT(12,12)
      L=DSQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)
      VO=A*L
      LAM=(X2-X1)/L
      MU=(Y2-Y1)/L
      NU=(Z2-Z1)/L
      SMAT(1,1)=LAM**2
      SMAT(2,2)=MU**2
      SMAT(3,3)=NU**2
      SMAT(4,4)=SMAT(1,1)
      SMAT(5,5)=SMAT(2,2)
      SMAT(6,6)=SMAT(3,3)
      SMAT(1,2)=MU*LAM
      SMAT(1,3)=NU*LAM
      SMAT(1,4)=-(LAM)**2
      SMAT(1,5)=-(MU*LAM)
      SMAT(1,6)=-(NU*LAM)
      SMAT(2,3)=NU*MU
      SMAT(2,4)=-LAM*MU
      SMAT(2,5)=-(MU)**2
      SMAT(2,6)=-NU*MU
      SMAT(3,4)=-LAM*NU
      SMAT(3,5)=-MU*NU
      SMAT(3,6)=-(NU)**2
      SMAT(4,5)=MU*LAM
      SMAT(4,6)=LAM*NU
      SMAT(5,6)=MU*NU
      DO 100 I=1,6
      DO 100 J=I,6
  100 SMAT (J,I)=SMAT(I,J)
      DO 105 I=1,6
      DO 105 J=1,6
  105 SMAT(I,J)=(A*E)/L*SMAT(I,J)
      RETURN
      END
```

```
MEMBER NA.   UNPCK689
      SUBROUTINE UNPACK (NONE,NTWO,KK)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION KK(1)
      KK(1)=NONE/100
      KK(2)=(NONE-100*KK(1))/10
      KK(3)=NONE-100*KK(1)-10*KK(2)
      KK(4)=NTWO/100
      KK(5)=(NTWO-100*KK(4))/10
      KK(6)=NTWO-100*KK(4)-10*KK(5)
      RETURN
      END
```

APPENDIX I

LISTING OF PROGRAM TO IDENTIFY AREAS OF HIGH STRAIN ENERGY DENSITY FOR HOUSINGS (S-83)

```
MEMBER NAME  S83PLAT2
C     MEMBERNAME S83PLAT2
      REAL*8 HEADR(8),EST
      REAL*8 TREF,BEND,TYME,BLANK,XMDE
      REAL*8 TIME
      COMMON/NOUTU/IOUT
      COMMON/STSART/I2ID(200),DDUM(600),STE,STMU
      DIMENSION IDP(12853),JPLT(60)
      DIMENSION S2(12853)
      EQUIVALENCE (S2(1),REC2(1))
      COMMON IREC1(146),REC2(12853),SIGMA(650,8,8),S1(1803),
     1LOADCS,ITYPE,IOEF1,SIGMS(650,16),IDELEM(650),IXCLUD(50),NELEM,
     2IPLT(10,6),NEID,DPTH(650),ITIME
      COMMON/MODE/XMDE
      EQUIVALENCE (IDP(1),REC2(1))
      NAMELIST/DATAX/JPLT
      NAMELIST /OMIT/IXCLUD
      REAL*8 NAME(4),NAMEIT,GPL,FILES(5),FILEF(5)
      DATA GPL/'GPL     '/
      DATA FILES/8HOESB1    ,8HOESC1    ,8HOESC2    ,8HOES1     ,8HOES2     /
      DATA FILEF/8HOEFB1    ,8HOEFC1    ,8HOEFC2    ,8HOEF1     ,8HOEF2     /
      DATA HEADR/'NORMAL-X','NORMAL-Y','SHEAR-XY','MAJOR   ','MINOR   ',
     1'SHEAR   ','THETA   ','CASE    '/
      DATA EST/'EST     '/
      DATA JPLT/6,7,8,9,15,16,17,18,19,63,3*17,8,17,8,3*17,8,4*5,2*6,5,
     13*6,9,2*7,3,7,3*3,9,3,4*12,2*16,12,3*16,27,2*25,21,30,26,21,26,32,
     226/
      DATA BLANK/8H        /
      DATA BEND/8HBENDING /
C
C   THIS PROGRAM CALCULATES CENTERLINE STRESSES FOR NASTRAN PLATE ELEMEN
C   ONLY AND FORMULATES A MAX/MIN ARRAY FOR THE CENTERLINE STRESSES.  TH
C   GRAM REQUIRES THAT THE NASTRAN DATA BLOCKS OES1 AND OEF1 EXIST ON A
C   TAPE.  MULTIPLE SUBROUTINES ARE USED AND THEIR FUNCTIONS ARE AS FOLL
C        1)  READS  - READS STRESS DATA FROM OES1 AND COPIES IT INTO SIG
C                     ARRAY.
C        2)  READF  - READS PLATE FORCES FROM OEF1 FOR TYPE AND LOAD CAS
C                     FLAGGED FROM (1).  DATA IS LEFT IN WORKING VECTOR
C        3)  STRESS - CALCULATES CENTERLINE STRESSES AND ASSOCIATED PRIN
C                     STRESSES.
C        4)  LOADMM - LOADS THE MAX/MIN TABLE AS APPROPIATE.
C        5)  RITE   - WRITES HEADER FOR STRESS OUTPUT
C        6)  PRTOVR - PRINTS OVER MAX/MIN TERMS IN SIGMA ARRAY.
C
C
C   READ IN
C        1)  ELEMENT FORCE FLAG (IOEF1).  IF IOEF1 LT 0, NO ELEMENT FORC
C            WILL BE PROCESSED.  IF IOEF1 EQ -1, THE FIRST FIBRE DISTANC
C            FROM THE PROPERTY CARD WILL BE USED.  OTHERWISE, THE SECOND
C            WILL BE USED.
C        2)  NAMES OF NASTRAN DATA FILES (AND UNIT NUMBER ON WHICH FILE
C            BE WRITTEN)  CONTAINING STRESSES AND FORCES IN THE ORDER TH
C            APPEAR ON CHECKPOINT TAPE.
C        3)  THE LOAD CASE ID NUMBERS TO BE EXCLUDED FROM CONSIDERATION
C            THE MAX/MIN ROUTINE. VARIABLE IS IXCLUD WITH UP TO 50 EXCLU
C            POSSIBLE.  NAMELIST INPUT IS USED.  LIST IS TERMINATED WITH
C
C     WRITE(6,DATAX)
      READ(5,20)(NAME(I),I=1,4),TREF,IOEF1,TYME,XMDE,STE,STMU
 20   FORMAT(5(A8),I8,A8,A8,2E8.0)
C  MEMBRANE THICKNESS
      TFLAG=-1.
C  BENDING THICKNESS
      IF(TREF.EQ.BEND) TFLAG=1.
C  SET TIMER FLAG
```

```
      ITIME=-1
      IF(TYME.NE.BLANK) ITIME=1
C  ZERO TIMER
      IF(ITIME.GT.0) CALL CLOCK(0,TIME)
      DO 31 I=1,50
   31 IXCLUD(I)=0
      DO 32 J=1,6
      DO 32 I=1,10
      IA=10*J+I-10
   32 IPLT(I,J)=JPLT(IA)
      READ(5,OMIT,END=30,ERR=30)
   30 CONTINUE
C   INITIALIZE ARRAYS
      DO 1 M=1,650
      IDELEM(M)=0
      DO 1 MA=1,8
      DO 1 MB=1,8
    1 SIGMA(M,MA,MB)=0.0
      DO 2 M=1,650
      SIGMA(M,1,1)=-1.E60
      SIGMA(M,2,2)=-1.E60
      SIGMA(M,1,3)=+1.E60
      SIGMA(M,2,4)=+1.E60
      SIGMA(M,4,6)=-1.E60
      SIGMA(M,5,7)=+1.E60
    2 CONTINUE
      ITYPE = 0
C
C   THE CHECKPOINT TAPE IS ASSUMED TO BE ON UNIT 10.
C
C   COPY DATA BLOCKS ONTO SCRATCH TAPES
C
      K=1
      DO 3000 I=1,4
      IOUT=I
      REWIND I
      NAMEIT=NAME(I)
      IF(NAMEIT.EQ.BLANK)GO TO 3004
      CALL XFETCH(1803,NAMEIT,REC2,IX,IY,12853,K,S1,10,S2,ISUC,1,4)
      IF(ISUC.GT.0) GO TO 3002
      WRITE(6,3001) NAMEIT
 3001 FORMAT(1H1,20(/),1X,120('*'),/,10X,'TROUBLE READING ',A8,' FROM RE
     1START TAPE--FATAL ERROR',//,1X,120('*'))
      STOP
 3002 K=0
      REWIND I
 3000 CONTINUE
 3004 DO 3010 I=1,4
      IF(NAME(I).EQ.EST) KEST=I
      IF(NAME(I).EQ.GPL) KGPL=I
      DO 3010 J=1,5
      IF(NAME(I).EQ.FILES(J)) KSTR=I
      IF(NAME(I).EQ.FILEF(J)) KFRC=I
 3010 CONTINUE
      CALL SORTID(KSTR)
      CALL LODTHK(KEST ,IFLAG)
C
C   START READING DATA BLOCKS FROM SCRATCH TAPES.
C
   50 CALL READS(KSTR ,&51)
      IF(ITYPE.EQ.9.OR.ITYPE.EQ.16.OR.ITYPE.EQ.63) GO TO 61
      IF(ITYPE.GT.0) GO TO 10
      WRITE(6,11)
   11 FORMAT(5X,'TROUBLE WITH OES1')
      STOP
   10 CONTINUE
      IF(IOEF1.GE.0) CALL READF(KFRC )
C
C   PLATE STRESSES ARE NOW CONTAINED IN THE ARRAY SIGMS, WITH ID CONTAIN
C   IDELEM.  PLATE FORCES AND ID'S ARE CONTAINED IN TEMPORARY VECTOR REC
```

```
C
C
C    CALL THE ROUTINE STRESS FOR CALCULATING CENTERLINE STRESSES AND PERF
C    MAX/MIN SEARCH.
C
   61 CALL STRESS
      GO TO 50
   51 J=0
C    PRINT TABLE OF MAX/MIN DATA
      DO 60 M=1,NELEM
      J=J+1
      IF(J.EQ.1) WRITE(6,70)
   70 FORMAT(1H1,//,20X,'S U M M A R Y   O F   M A X I M U M   A N D  M
     1 I N I M U M   S T R E S S E S',/,40X,'( READ DATA SETS BY COLUMNS
     2 )')
      IF(J.EQ.3) J=0
      WRITE(6,71) IDELEM(M),DPTH(M)
   71 FORMAT( //,45X,'E L E M E N T   N O.',I8,/,50X,'(THICKNESS=',F6.4,
     1')',//,17X,'MAXIMUM TENSION',13X,'MAXIMUM COMPRESSION',33X,
     2'PRINCIPAL STRESSES',//,13X,'NORMAL-X',7X,'NORMAL-Y',7X,'NORMAL-X'
     3,7X,'NORMAL-Y',7X,'SHEAR-XY',9X,'MAJOR',10X,'MINOR',8X,'SHEAR',/)
      DO 72 MI=1,8
      IF(MI.GT.6) GO TO 74
      WRITE(6,73) HEADR(MI),(SIGMA(M,MI,MJ),MJ=1,8)
   73 FORMAT(1X,A8,8(2X,E13.7))
C    PRINT OVER DIAGONAL TERMS
      CALL PRTOVR(M,MI)
      GO TO 72
   74 IF(MI.EQ.7) WRITE(6,75) HEADR(MI),(SIGMA(M,MI,MJ),MJ=1,8)
   75 FORMAT(1X,A8,8(6X,F9.4))
      IF(MI.EQ.8) WRITE(6,76) HEADR(MI),(SIGMA(M,MI,MJ),MJ=1,8)
   76 FORMAT(1X,A8,8(8X,F7.0))
   72 CONTINUE
   60 CONTINUE
      CALL ELCON(KEST,KGPL)
      STOP
      END
      SUBROUTINE SORTID(IUNIT)
      REAL*8 TIME,XMODE,XMDE
      REAL*8 FILEN
      COMMON/NOUTU/IOUT
      COMMON IREC1(146),REC2(12853),SIGMA(650,8,8),S1(1803),
     1LOADCS,ITYPE,IOEF1,SIGMS(650,16),IDELEM(650),IXCLUD(50),NELEM,
     2IPLT(10,6),NFID,DPTH(650),ITIME
      COMMON/MODE/XMDE
      DATA XMODE/'MODE    '/
      DIMENSION IDP(12853)
      EQUIVALENCE (IDP(1),REC2(1))
      IF(ITIME.LT.0) GO TO 4001
      CALL CLOCK(1,TIME)
      TIM=(TIME*3.6D3)
      WRITE(6,1000) TIM
 4001 CONTINUE
 1000 FORMAT(1X,'ENTERING SORTID',F16.8)
C   THIS SUBROUTINE SCANS THE STRESS TABLE, BUILDS IDELEM ARRAY, AND THEN
C   IT IN ASCENDING ORDER.
      NELEM = 0
      KASFLG= -1
   99 READ(IUNIT,END=101)FILEN,JJ,L
      READ(IUNIT,END=101)(IREC1(II),II=1,L)
      READ(IUNIT,END=101)FILEN,JJ,L
      READ(IUNIT,END=101)( REC2(II),II=1,L)
C   CHECK FOR PLATE ELEMENT
      ITYPE=IREC1(3)
      DO 10 I=1,10
      IF(ITYPE.EQ.IPLT   (I,1)) GO TO 98
   10 CONTINUE
      GO TO 99
   98 LOADCS=IREC1(4)
      IF(XMDE.EQ.XMODE)LOADCS=IREC1(5)
```

```
C   SET FIRST LOAD CASE NUMBER
       IF(KASFLG.EQ.-1) LOAD1=LOADCS
       IF(LOADCS.GT.LOAD1) GO TO 101
       KASFLG=1
C   SET NUMBER OF ELEMENTS
       NEID=L/IPLT   (I,2)
       DO 90 J=1,NEID
       K=NELEM+J
       JA=(J-1)*IPLT   (I,2)+1
C   FILL UP ELEMENT ID ARRAY
   90  IDELEM(K)=(IDP(JA)-1)/10
       NELEM=NELEM+NEID
       GO TO 99
  101  CONTINUE
C   SORT THE ID'S
       KX=NELEM
   20  KN1=1
   21  KO=KX-1
       LO=0
       DO 30 N=1,KO
       GO TO (22,23),KN1
   22  I=N
       GO TO 24
   23  I=KX-N
   24  CONTINUE
       IF(IDELEM(I)-IDELEM(I+1)) 30,30,25
   25  NARY=IDELEM(I)
       IDELEM(I)=IDELEM(I+1)
       IDELEM(I+1)=NARY
       LO=1
   30  CONTINUE
       IF(LO) 40,40,35
   35  IF(KN1-1) 45,45,20
   45  KN1=2
       GO TO 21
   40  CONTINUE
       REWIND IUNIT
       ITYPE=0
       RETURN
  100  WRITE(6,102)
  102  FORMAT(1H1,10(/),1X,'TROUPLE IN SUBROUTINE SORTID---EXIT.')
       STOP
       END
       SUBROUTINE LODTHK(IUNIT,TFLAG)
       REAL*8 TIME
       DIMENSION IDP(12853)
       COMMON IREC1(146),REC2(12653),SIGMA(650,8,8),S1(1803),
      1LOADCS,ITYPE,IOEF1,SIGMS(650,16),IDELEM(650),IXCLUD(50),NELEM,
      2IPLT(10,6),NEID,DPTH(650),ITIME
       EQUIVALENCE (IDP(1),REC2(1))
       REAL*8 FILE
       COMMON/NOUTU/IOUT
       IOUT=IUNIT
       IF(ITIME.LT.0) GO TO 4001
       CALL CLOCK(1,TIME)
       TIM=(TIME*3.6D3)
       WRITE(6,1000) TIM
 4001  CONTINUE
 1000  FORMAT(1X,'ENTERING LODTHK',F16.8)
C   THIS SUBROUTINE LOADS THE PLATE THICKNESSES
   42  READ(IUNIT,END=100) FILE,JJ,L
       IF(L.LE.6) RETURN
       READ(IUNIT,END=100) (REC2(I),I=1,L)
       J=IDP(1)
       DO 43 I=1,10
       IF(J.EQ.IPLT(I,1)) GO TO 50
   43  CONTINUE
       GO TO 42
   50  CONTINUE
       IECT=  IPLT(I,3)
```

```
      IEPT=  IPLT(I,4)
      IRGPDT=IPLT(I,5)
      ITOT=  IPLT(I,6)
C  SET NUMBER OF ELEMENTS
      NEID=(L-1)/ITOT
C  LOOP THROUGH ELEMENTS
      DO 10 IA=1,NEID
      IB=(IA-1)*ITOT+2
      IEID=IDP(IB)
      IF(IEID.LT.IDELEM(1)) GO TO 10
      IF(IEID.GT.IDELEM(NELEM)) GO TO 42
      DO 20 IC=1,NELEM
      IF(IEID.NE.IDELEM(IC)) GO TO 20
      DPTH(IC)=REC2(IB+IECT+1)
      IF(J.EQ.7.OR.J.EQ.8.OR.J.EQ.15) DPTH(IC)=REC2(IB+IECT+3)
      IF(YFLAG.LE.0.) GO TO 20
      IF(J.EQ.6.OR.J.EQ.19) DPTH(IC)=REC2(IB+IECT+5)
   20 CONTINUE
   10 CONTINUE
      GO TO 42
  100 RETURN
      END
      SUBROUTINE READS(IUNIT,*)
      REAL*8 FILEN,XMODE,XMDE
      REAL*8 TIME
      COMMON/NOUT/IOUT
      DIMENSION IDP(12853)
      COMMON IREC1(146),REC2(12853),SIGMA(650,8,8),S1(1803),
     1LOADCS,ITYPE,IDEF),SIGMS(650,16),IDELEM(650),IXCLUD(50),NELEM,
     2IPLT(10,6),NCID,DPTH(650),ITIME
      COMMON/MODE/XMDE
      DATA XMODE/'MODE    '/
      EQUIVALENCE (IDP(1),REC2(1))
      IOUT=IUNIT
      IF(ITIME.LT.0) GO TO 4001
      CALL CLOCK(1,TIME)
      TIM=(TIME*3.6D3)
      WRITE(6,1000) TIM
 4001 CONTINUE
 1000 FORMAT(1X,'ENTERING READS',F16.8)
C   BEGIN READING DATA FROM OESI
C
   99 CONTINUE
      READ(IUNIT,END=100) FILEN,JJ,L
      IF(L.LT.100) RETURN 1
C     WRITE(6,12) FILEN,JJ,L
   12 FORMAT(1X,A8,2(4X,I8))
      READ(IUNIT,END=100) (IREC1(II),II=1,L)
C     WRITE(6,11) (IREC1(II),II=1,L)
   11 FORMAT(1X,15I8)
      READ(IUNIT,END=100) FILEN,JJ,L
      READ(IUNIT,END=100) (REC2(II),II=1,L)
C     WRITE(6,201) (REC2(II),II=1,L)
C 201 FORMAT(1X,8(F14.4,1X))
C  CHECK FOR PLATE ELEMENTS
      DO 10 I=1,10
      IF(IREC1(3).EQ.IPLT(I,1)) GO TO 98
   10 CONTINUE
      GO TO 99
C
C  PLATE ELEMENT FOUND, PROCEED
C
 98   LOADCS=IREC1(4)
      IF(XMDE.EQ.XMODE)LOADCS=IREC1(5)
      ITYPE=IREC1(3)
C   SET NUMBER OF ELEMENTS IDS
      NEID=L/IPLT(I,2)
      DO 90 J=1,NEID
      JA=(J-1)*IPLT(I,2)+1
      ID=(IDP(JA)-1)/10
```

```
      DO 91 K=1,NELEM
      IF(ID.EQ.IDELEM(K)) GO TO 92
   91 CONTINUE
   92 IEND=IPLT(I,2)-1
C   LOOP THROUGH ELEMENTS AND STORE IN SIGMS ARRAY.
C
   86 DO 89 JD=1,IEND
   89 SIGMS(K ,JD)=REC2(JA+JD)
C     WRITE(6,200) (SIGMS(K ,M),M=1,16)
   90 CONTINUE
      RETURN
  200 FORMAT(1X,8(F14.4,1X),/,1X,8(F14.4,1X))
  100 CONTINUE
      IF(IOEF1.LT.0) RETURN 1
      RETURN
      STOP
      END
      SUBROUTINE READF(IUNIT)
      REAL*8 FILEN
      REAL*8 TIME
      DIMENSION IOP(12853)
      COMMON IREC1(196),REC2(12853),SIGMA(650,8,8),S1(1803),
     1LOADCS,ITYPE,IOEF1,SIGMS(650,16),IDELEM(650),IXCLUD(50),NELEM,
     2IPLT(10,6),NFID,DPTH(650),ITIME
      EQUIVALENCE (IOP(1),REC2(1))
      COMMON/NOUTU/IOUT
      IF(ITIME.LT.0) GO TO 4001
      CALL CLOCK(1,TIME)
      TIM=(TIME*3.6D3)
      WRITE(6,1000) TIM
 4001 CONTINUE
 1000 FORMAT(1X,'ENTERING READF',F16.8)
      IOUT=IUNIT
      NREWND=-1
C
C   COPY OEF1 FROM TAPE
C   NREWND= NUMBER OF REWINDS
   98 NREWND=NREWND+1
   99 CONTINUE
C     WRITE(6,1) ITYPE,LOADCS
    1 FORMAT(1X,20('*'),2I8)
      READ(IUNIT,END=100) FILEN,JJ,L
C     WRITE(6,12) FILEN,JJ,L
   12 FORMAT(1X,A8,2(4X,I8))
      READ(IUNIT,END=100) (IREC1(II),II=1,L)
C     WRITE(6,11) (IREC1(II),II=1,L)
   11 FORMAT(1X,15I8)
      READ(IUNIT,END=100) FILEN,JJ,L
      READ(IUNIT,END=100) (REC2(II),II=1,L)
      IF(NREWND.GT.5) GO TO 110
      DO 10 I=1,10
      IF(IREC1(3).EQ.IPLT(I,1)) GO TO 102
   10 CONTINUE
      GO TO 99
C
C  PLATE FOUND, PROCEED
C
  102 IF(IREC1(3).NE.ITYPE.OR.IREC1(4).NE.LOADCS) GO TO 99
C   ELEMENT TYPE AND LOAD CASE MATCH THOSE FROM 'READS' -- PROCEED.
      RETURN
  100 REWIND IUNIT
      GO TO 98
  110 WRITE(6,111)
  111 FORMAT(1H1,20(/),120('*'),//,2X,'THE UNIT CONTAINING OEF1 HAS BEEN
     1 REWOUND 5 TIMES -- VERIFY THAT PLATE FORCES ARE ON TAPE',//,120('
     2*'))
      STOP
      END
      SUBROUTINE STRESS
      REAL*8 TIME
      DIMENSION IOP(12853)
```

```
      COMMON IREC1(146),REC2(12853),SIGMA(650,8,8),SI(1803),
     1LOADCS,ITYPE,IOEF1,SIGMS(650,16),IDELEM(650),IXCLUD(50),NELEM,
     2IPLT(10,6),NEID,DPTH(650),ITIME
      EQUIVALENCE (IDP(1),REC2(1))
      COMMON/STSART/IZIO(200),ZSIG1(200),ZSIG2(200),STDEN(200),STE,STMU
C   LOOP THROUGH ELEMENTS
      IF(ITIME.LT.0) GO TO 4001
      CALL CLOCK(1,TIME)
      TIM=(TIME*3.6D3)
      WRITE(6,1000) TIM
 4001 CONTINUE
 1000 FORMAT(1X,'ENTERING STRESS',F16.8)
      JC=0
      J=1
C  SET RECORD LENGTH
C
C  IF ITYPE .EQ. 9,16,63, LNG=8
      LNG=8
      IF(ITYPE.EQ.9.OR.ITYPE.EQ.16.OR.ITYPE.EQ.63) GO TO 3
C  IF IOEF1 .GE. 0, LNG=6
      LNG=6
      IF(IOEF1.GE.0) GO TO 3
C  IF IOEF1 .LT. 0, LNG=17
      LNG=17
    3 DO 10 L=1,NEID
      KK=(L-1)*LNG+1
      ID=(IDP(KK)-1)/10
      DO 91 I=1,NELEM
      IF(ID.EQ.IDELEM(I)) GO TO 92
   91 CONTINUE
      WRITE(6,70) ID
   70 FORMAT(1H1,20(/),120('*'),//,'NO ELEMENT HAS BEEN FOUND IN OEF1 FO
     1R THE CURRENT LOAD CASE AND ELEMENT',I5,'.  THIS ELEMENT IS IGNORE
     2D.',//,120('*'))
      STOP
      GO TO 10
   92 CONTINUE
      JC=JC+1
      IF(JC.EQ.1) CALL RITE(ITYPE,LOADCS)
      IF(JC.EQ.50) JC=0
      IF(LNG.EQ.8) GO TO 99
      IF(IOEF1.LT.0) GO TO 35
C   CHECK FOR ZERO FIBRE DISTANCE.
      J=1
      IF(SIGMS(I,J).EQ.0.) GO TO 99
      J=9
      IF(SIGMS(I,J).EQ.0.) GO TO 99
      J=1
      IF(SIGMS(I,1).NE.(-SIGMS(I,9))) GO TO 5
C   DATA IS FOUND FOR EQUAL DISTANCES FROM CENTERLINE, AVERAGE DATA.
      SIGMS(I,J+1)=(SIGMS(I,J+1)+SIGMS(I,J+9))/2.
      SIGMS(I,J+2)=(SIGMS(I,J+1)+SIGMS(I,J+10))/2.
      GO TO 41
    5 CONTINUE
C   READ NONZERO FIBRE DISTANCE
      FIBDIS=AMAX1(SIGMS(I,1),SIGMS(I,9))
      IF(SIGMS(I,9).GT.SIGMS(I,1)) J=9
      TSQ06=(2.*FIBDIS)**2/6.
   40 CONTINUE
C  IF YOU'RE HERE, YOU FOUND AN ELEMENT MATCH.
      BENSTS=-REC2(KK+1)/TSQ06
      IF(FIBDIS.LT.0.) BENSTS=-BENSTS
C  STRESS IS POSITIVE IF TENSION.
      SIGMS(I,J+1)=SIGMS(I,J+1)-BENSTS
      BENSTS=-REC2(KK+2)/TSQ06
      IF(FIBDIS.LT.0.) BENSTS=-BENSTS
      SIGMS(I,J+2)=SIGMS(I,J+2)-BENSTS
C   PRINCIPAL STRESSES
   41 SIG1=(SIGMS(I,J+1)+SIGMS(I,J+2))/2.
      SIG2=SQRT((SIGMS(I,J+1)-SIGMS(I,J+2))**2/4.+SIGMS(I,J+3)**2)
```

```
      SIG3=SIG1+SIG2
      SIG4=SIG1-SIG2
      SIGMS(I,J+5)=AMAX1(SIG3,SIG4)
      SIGMS(I,J+6)=AMIN1(SIG3,SIG4)
      SIGMS(I,J+4)=(ATAN(2.*SIGMS(I,J+3)/(SIGMS(I,J+2)-SIGMS(I,J+1))))
     1/2.*180./3.1416
      SIGMS(I,J+7)=ABS(SIG2)
      IF(SIGMS(I,J+4).GT.45..OR.SIGMS(I,J+4).LT.-45.)
     *SIGMS(I,J+4)=ABS(SIGMS(I,J+4))-90.
   35 IF(IOEF1.LT.-1) J=9
   99 JP1=J+1
      JP7=J+7
      WRITE(6,98) ID,(SIGMS(I,IJ),IJ=JP1,JP7),DPTH(I)
   98 FORMAT(I8,6X,3(E13.7,2X),F12.4,   3(2X,E13.7),7X,F6.4)
      IF(ITYPE.EQ.6)GO TO100
      IF(ITYPE.EQ.19)GO TO100
      IF(ITYPE.EQ.17)GO TO 100
      IF(ITYPE.EQ.18)GO TO 100
      GO TO 200
  100 IZID(L)=ID
      ZSIG1(L)=SIGMS(I,JP7-2)
      ZSIG2(L)=SIGMS(I,JP7-1)
      STDEN(L)=.5*(ZSIG1(L)**2+ZSIG2(L)**2-2.*STMU*ZSIG1(L)*ZSIG2(L))/
     1STE
C   SKIP MAX/MIN IF LOAD CASE EXCLUDED
      ICLUD=0
  200 DO 79 IXY=1,50
      IF(IXCLUD(IXY).EQ.0) GO TO 78
      IF(LOADCS.EQ.IXCLUD(IXY))ICLUD=1
      IF(LOADCS.EQ.IXCLUD(IXY)) GO TO 10
   79 CONTINUE
   78 CONTINUE
C  NOW FILL UP MAX/MIN ARRAY.
      IF(SIGMS(I,J+1).LE.SIGMA(I,1,1)) GO TO 80
      CALL LOADMM(I,1,J,I)
   80 IF(SIGMS(I,J+1).GT.SIGMA(I,1,3)) GO TO 81
      CALL LOADMM(I,3,J,I)
   81 IF(SIGMS(I,J+2).LE.SIGMA(I,2,2)) GO TO 82
      CALL LOADMM(I,2,J,I)
   82 IF(SIGMS(I,J+2).GT.SIGMA(I,2,4)) GO TO 83
      CALL LOADMM(I,4,J,I)
   83 IF(ABS(SIGMS(I,J+3)).LE.ABS(SIGMA(I,3,5))) GO TO 84
      CALL LOADMM(I,5,J,I)
   84 IF(SIGMS(I,J+4).LE.SIGMA(I,4,6)) GO TO 85
      CALL LOADMM(I,6,J,I)
   85 IF(SIGMS(I,J+5).GT.SIGMA(I,5,7)) GO TO 86
      CALL LOADMM(I,7,J,I)
   86 IF(ABS(SIGMS(I,J+6)).LE.ABS(SIGMA(I,6,8))) GO TO 10
      CALL LOADMM(I,8,J,I)
   10 CONTINUE
      IF(ICLUD.EQ.1)GO TO 300
      CALL STSORT
  300 IF(ITIME.LT.0)GO TO 4002
      CALL CLOCK(1,TIME)
      TIM=(TIME*3.6D3)
      WRITE(6,1001) TIM
 4002 CONTINUE
 1001 FORMAT(1X,'LEAVING STRESS',F16.8)
      RETURN
      STOP
      END
      SUBROUTINE LOADMM(JDELEM,ICOL,J,KDELEM)
      DIMENSION IOP(12853)
      COMMON IREC1(146),REC2(12853),SIGMA(650,8,8),SI(1803),
     1LOADCS,ITYPE,IOFF1,SIGMS(650,16),IDELEM(650),IXCLUD(50),NELEM,
     2IPLT(10,6),NFID,DPTH(650),ITIME
      EQUIVALENCE (IOP(1),REC2(1))
C
C    THIS SUBROUTINE LOADS THE ICOL COLUMN OF THE MAX/MIN ARRAY FOR THE
C    JDELEM ELEMENT.
```

```
C
      SIGMA(KDELEM,1,ICOL)=SIGMS(JDELEM,J+1)
      SIGMA(KDELEM,2,ICOL)=SIGMS(JDELEM,J+2)
      SIGMA(KDELEM,3,ICOL)=SIGMS(JDELEM,J+3)
      SIGMA(KDELEM,4,ICOL)=SIGMS(JDELEM,J+5)
      SIGMA(KDELEM,5,ICOL)=SIGMS(JDELEM,J+6)
      SIGMA(KDELEM,6,ICOL)=SIGMS(JDELEM,J+7)
      SIGMA(KDELEM,7,ICOL)=SIGMS(JDELEM,J+4)
      SIGMA(KDELEM,8,ICOL)=LOADCS
      RETURN
      STOP
      END
      SUBROUTINE PRTOVR(I,J)
      DIMENSION IDP(12853)
      COMMON IREC1(146),REC2(12853),SIGMA(650,8,8),S1(1803),
     1LOADCS,ITYPE,IOEF1,SIGMS(650,16),IDELEM(650),IXCLUD(50),NELEM,
     2IPLT(10,6),NEID,OPTH(650),ITIME
      EQUIVALENCE (IDP(1),REC2(1))
      GO TO(10,20,30,40,50,60),J
   10 WRITE(6,11) SIGMA(I,J,1),SIGMA(I,J,3)
      WRITE(6,11) SIGMA(I,J,1),SIGMA(I,J,3)
   11 FORMAT(1H+,10X,E13.7,17X,E13.7)
      RETURN
   20 WRITE(6,21) SIGMA(I,J,2),SIGMA(I,J,4)
      WRITE(6,21) SIGMA(I,J,2),SIGMA(I,J,4)
   21 FORMAT(1H+,25X,E13.7,17X,E13.7)
      RETURN
   30 WRITE(6,31) SIGMA(I,J,5)
      WRITE(6,31) SIGMA(I,J,5)
   31 FORMAT(1H+,70X,E13.7)
      RETURN
   40 WRITE(6,41) SIGMA(I,J,6)
      WRITE(6,41) SIGMA(I,J,6)
   41 FORMAT(1H+,85X,E13.7)
      RETURN
   50 WRITE(6,51) SIGMA(I,J,7)
      WRITE(6,51) SIGMA(I,J,7)
   51 FORMAT(1H+,100X,E13.7)
      RETURN
   60 WRITE(6,61) SIGMA(I,J,8)
      WRITE(6,61) SIGMA(I,J,8)
   61 FORMAT(1H+,115X,E13.7)
      RETURN
      STOP
      END
      SUBROUTINE RITE(ITYPE,LOADCS)
      WRITE(6,10)
   10 FORMAT(1H1,//,30X,'S T R E S S E S   I N   ')
      IF(ITYPE.EQ. 6) WRITE(6, 6)
      IF(ITYPE.EQ. 7) WRITE(6, 7)
      IF(ITYPE.EQ. 8) WRITE(6, 8)
      IF(ITYPE.EQ. 9) WRITE(6, 9)
      IF(ITYPE.EQ.15) WRITE(6,15)
      IF(ITYPE.EQ.16) WRITE(6,16)
      IF(ITYPE.EQ.17) WRITE(6,17)
      IF(ITYPE.EQ.18) WRITE(6,18)
      IF(ITYPE.EQ.19) WRITE(6,19)
      IF(ITYPE.EQ.63) WRITE(6,63)
      WRITE(6,20) LOADCS
   20 FORMAT(1H+,70X,'P L A T E S   F O R   C A S E',I8,/)
    6 FORMAT(1H+,54X,'C T R I A 1  ')
    7 FORMAT(1H+,54X,'C T R B S C  ')
    8 FORMAT(1H+,54X,'C T R P L T  ')
    9 FORMAT(1H+,54X,'C T R M E M  ')
   15 FORMAT(1H+,54X,'C Q D P L T  ')
   16 FORMAT(1H+,54X,'C Q D M E M  ')
   17 FORMAT(1H+,54X,'C T R I A 2  ')
   18 FORMAT(1H+,54X,'C Q U A D 2  ')
   19 FORMAT(1H+,54X,'C Q U A D 1  ')
   63 FORMAT(1H+,54X,'C Q D M E M 2')
```

```
  1 WRITE(6,40)
 40 FORMAT(1X,'ELEMENT',10X,'STRESSES IN ELEMENT COORD SYSTEM',12X,
   1'PRINCIPAL',11X,'PRINCIPAL STRESSES')
    WRITE(6,50)
 50 FORMAT(3X,'ID.',10X,'NORMAL-X',7X,'NORMAL-Y',7X,'SHEAR-XY',5X,
   1'STRESS ANGLE',8X,'MAJOR',10X,'MINOR',10X,'SHEAR',7X,'THICKNESS',
   2//)
    RETURN
    END
    SUBROUTINE ELCON(KEST,KGPL)
    COMMON IREC1(146),REC2(12853),SIGMA(650,8,8),S1(1803),
   1LOADCS,ITYPE,IOEF1,SIGMS(650,16),IDELEM(650),IXCLUD(50),NELEM,
   2IPLT(10,6),NFID,DPTH(650),ITIME
    DIMENSION ICON(650,7),IGPL(1300),JEST(10),IDP(12853)
    REAL*8 FILEN,LNAME(12),NAME
    EQUIVALENCE (ICON(1,1),SIGMS(1,1)),(IGPL(1),SIGMS(1,8))
    EQUIVALENCE(IDP(1),REC2(1))
    DATA JEST/4*3,2*4,3,3*4/
    DATA LNAME/8HCTRIA1  ,8HCTRBSC  ,8HCTRPLT  ,8HCTRMEM  ,8HCQDPLT  ,
   18HCQDMEM  ,8HCTRIA2  ,8HCQUAD2  ,8HCQUAD1  ,8HCQDMEM2 /
    REWIND KEST
C   READ IN GPL TABLE
    READ(KGPL) FILEN,J,L
    READ(KGPL) (IGPL(I),I=1,L)
    NL=0
C   READ IN EST TABLE
 19 READ(KEST,END=100) FILEN,J,L
    READ(KEST,END=100) (REC2(I),I=1,L)
    IF(IDP(1).GT.10.AND.IDP(1).LT.15) GO TO 19
    IF(IDP(1).GT.19.AND.IDP(1).LT.31) GO TO 19
    IF(L.LE.6) GO TO 100
    LTYPE=IDP(1)
C   DETERMINE ELEMENT TYPE
    DO 20 IA=1,10
    IF(LTYPE.EQ.IPLT(IA,1)) GO TO 30
 20 CONTINUE
    GO TO 19
 30 CONTINUE
    LNG=IPLT(IA,6)
    NELEM=L/LNG
    NS=NL+1
    NL=NL+NELEM
    K=0
    DO 40 I=NS,NL
    K=K+1
    IB=(K-1)*LNG+2
    IC=JEST(IA)
    ICON(I,1)=IDP(IB)
    ICON(I,2)=IA
    ICON(I,3)=IC
    DO 41 J=1,IC
    INGPL=(IDP(IB+J)+5)/6
 41 ICON(I,J+3)=IGPL(INGPL)
 40 CONTINUE
    GO TO 19
C   CONNECTION ARRAY IS COMPLETE, SORT IT
100 CONTINUE
    NELEM=NL
    CALL SORTCX(ICON,650,7,1)
C     NOW PRINT IT
    M=0
    DO 50 I=1,NL
    IF(M.EQ.0) WRITE(6,51)
 51 FORMAT(1H1,/,40X,'E L E M E N T   C O N N E C T I O N   T A B L E'
   1,//,7X,'ELEMENT',8X,'ELEMENT',1X,4(8X,'GRID',3X),/,9X,'ID',12X,
   2'NAME',13X,'A',14X,'B',14X,'C',14X,'D',//)
    NAME=LNAME(ICON(I,2))
    IC=ICON(I,3)+3
    WRITE(6,60) ICON(I,1),NAME,(ICON(I,J),J=4,IC)
 60 FORMAT(7X,I6,9X,A8,2X,4(3X,I6,6X))
    M=M+1
```

I13

MEMBER NAME SA3PLAT2

```
      IF(M.GE.51) M=0
   50 CONTINUE
      RETURN
      END
      SUBROUTINE  SORTCX(ITEMP,L,M,ISEL)
      COMMON IREC1(140),REC2(12853),SIGMA(650,8,8),S1(1803),
     1LOADCS,ITYPE,IOCF1,SIGMS(650,16),IDELEM(650),IXCLUD(50),NELEM,
     2IPLT(10,6),NEIO,DPTH(650),ITIME
C
C
C        ITEMP IS THE ARRAY TO BE SORTED
C        L AND M ARE THE ADJUSTABLE DIMENSION VARIABLES FOR ITEMP
C
C
C        ISEL = THE NO. OF THE ROW OR COL. ON WHICH TO SORT
C
C
      DIMENSION  ITEMP(L,M),NARY(650)
      EQUIVALENCE (NARY(1),IDELEM(1))
      LL = NELEM
      MM = M
      IF(L.EQ.1.AND.M.EQ.1)GO TO 71
      KX = LL
   20 KN1 = 1
   21 KO = KX - 1
      LO = 0
      DO 30 N = 1,KO
      GO TO (22,23),KN1
   22 I = N
      GO TO 24
   23 I = KX - N
   24 CONTINUE
      IF(ITEMP(I,ISEL)-ITEMP(I+1,ISEL)) 30,30,25
   25 DO 60 II = 1,MM
      NARY(  II) = ITEMP(I,II)
      ITEMP(I,II)= ITEMP(I+1,II)
      ITEMP(I+1,II) = NARY(  II)
   60 CONTINUE
      LO =1
   30 CONTINUE
      IF(LO)40,40,35
   35 IF(KN1-1)45,45,20
   45 KN1 = 2
      GO TO 21
   40 CONTINUE
   70 CONTINUE
   71 RETURN
      END
```